

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Alternativas para a Interoperabilidade entre Sistemas de Informação Universitários

Sérgio Sobral Nunes

Licenciado em Engenharia Informática e Computação pela
Faculdade de Engenharia da Universidade do Porto

Dissertação submetida para satisfação parcial dos
requisitos do grau de mestre em
Gestão de Informação

Dissertação realizada sob a supervisão do
Professor Doutor Gabriel David,
do Departamento de Engenharia Electrotécnica e de Computadores
da Faculdade de Engenharia da Universidade do Porto

Porto, Julho de 2004

Resumo

Os sistemas de informação são peças centrais e incontornáveis nas organizações actuais. Nas instituições académicas, podem ser encontrados em áreas tão diversas como a gestão financeira da organização, o apoio à actividade lectiva ou a gestão de recursos humanos.

A evolução das tecnologias e infraestruturas associadas às redes de computadores veio colocar novos desafios às organizações. Os sistemas centralizados têm sido progressivamente substituídos por sistemas de informação distribuídos, compostos por múltiplos computadores ligados em rede. Por outro lado, de um ponto de vista organizacional, há uma crescente necessidade de aceder a informação consolidada e agregada, proveniente de diferentes fontes. No caso particular das instituições académicas, caracterizadas pela existência de múltiplos centros de poder, esta questão assume uma maior preponderância.

Assim, devido a factores de índole tecnológica e organizacional, a integração de informação constitui um dos mais importantes desafios actuais. Nesta dissertação, procede-se a um estudo e a uma avaliação das principais alternativas para a interoperabilidade entre sistemas de informação universitários.

Numa primeira fase, é abordado o conceito de Sistema de Informação Universitário e, através da análise das funcionalidades mais comuns, é feita uma breve caracterização destes sistemas. Aprofunda-se depois o conceito de interoperabilidade, atribuindo-se maior destaque à vertente tecnológica. É feita uma revisão das principais tecnologias e soluções associadas à construção de sistemas de informação distribuídos.

Em seguida, é apresentado o problema concreto de interoperabilidade entre sistemas de informação universitários, tendo subjacente como caso de estudo o cenário da Universidade do Porto. É feita uma avaliação dos padrões de interacção mais comuns e o problema é estruturado em dois eixos: horizontal e vertical. Com base na descrição e caracterização destes eixos, são identificados quatro padrões de interacção: partilha, agregação, concentração e difusão de informação. Cada um destes padrões dá origem a um caso de estudo concreto, que é descrito e sistematizado com a compilação de requisitos detalhados.

Propõem-se quatro arquitecturas distintas para cada um dos padrões genéricos. Depois de descritas e justificadas as opções tomadas, é feita uma breve contextualização para cada caso de estudo particular. Este trabalho teve como resultado a produção duma especificação de soluções concretas para a representação dos dados e para a implementação dos mecanismos de interacção.

No final, foram implementados vários protótipos que permitiram validar, numa primeira fase e informalmente, as soluções propostas.

Abstract

Information systems are fundamental pieces in today's organizations. In academic institutions, they can be found in areas as diverse as financial management of the organization, lecture support or human resources management.

The evolution of technologies and infrastructures in the field of computer networks has brought new challenges to the organizations. Centralized systems are being replaced by distributed ones, built on top of multiple connected computers. On the other hand, from an organizational point of view, there is a greater need to access consolidated and aggregated information produced by different sources. In the case of academic institutions, where multiple power centers exist, this is a bigger issue.

Thus, due to technological and organizational issues, information integration is one of the most important current challenges. In this dissertation, a study and evaluation of the major alternatives for the interoperability between university information systems is conducted.

First, the concept of University Information System is presented. A brief characterization of these systems is made based on the analyses of the most common features. Then, the concept of interoperability is detailed, with a focus on the technological aspects. The major technologies and solutions related to distributed systems are reviewed here.

Following this, the problem of interoperability between university information systems is presented having as an example the concrete case of the Porto University. An evaluation of the more common interaction patterns is made and the problem is structured in two axis: horizontal and vertical. Based on the description and characterization of these axis, four patterns are identified: share, aggregation, concentration and diffusion of information. Each of these patterns results in a study case, which is described and systematized with the compilation of detailed requirements.

Four distinct architectures are proposed for each of the generic patterns. After being described and justified the options, a brief contextualization is made for each particular case. This work resulted in the specification of concrete solutions for the representation of data and the implementation of interaction mechanisms.

In the end, several prototypes were implemented that allowed to validate the proposed solutions.

Entities should not be multiplied beyond necessity,
William of Ockham (1285-1347/49)

Agradecimentos

Ao professor Gabriel David por todo o apoio e dedicação com que acompanhou o trabalho.

À Carla e à minha mãe pelo apoio, pela disponibilidade constante e pelas críticas construtivas que foram tecendo sobre o meu trabalho.

Ao meu pai, aos meus irmãos, aos meus avós e ao Manuel pelo incentivo.

Aos meus amigos também pelo incentivo e, muito em particular, por toda a paciência demonstrada durante este (longo) período.

Sumário

1	Introdução	19
1.1	Enquadramento e Motivação	20
1.2	Organização da Dissertação	20
2	Sistemas de Informação Universitários	23
2.1	As Organizações Universitárias	24
2.2	Os Sistemas de Informação nas Universidades	25
2.3	Funcionalidades dos SIU	26
2.4	Troca de Informação	30
2.4.1	Definição	30
2.4.2	Iniciativas Existentes no Contexto Académico	31
3	Tecnologias de Apoio à Interoperabilidade	37
3.1	Níveis de Interoperabilidade	38
3.2	Arquitecturas Distribuídas	39
3.3	Modelos de Programação	40
3.4	Sincronismo	41
3.4.1	Interacção Síncrona	41
3.4.2	Interacção Assíncrona	42
3.5	Colocação dos Dados	42
3.5.1	Mover a Interrogação	42
3.5.2	Mover os Dados	43
3.5.3	Solução Híbrida	43
3.6	Consistência dos Dados	44
3.6.1	Protocolos	44
3.6.2	Propagação	45
3.6.3	Iniciativa	46
3.7	Entrega dos Dados	47
3.7.1	Protocolos	48
3.7.2	Frequência	48
3.7.3	Modos	48
3.7.4	Classificação dos Esquemas	48
3.8	Revisão das Tecnologias	48
3.8.1	EDI	49
3.8.2	Componentes	50
3.8.3	Serviços Web	51

4	Definição do Problema	53
4.1	Apresentação do Contexto	54
4.2	Eixos de Interoperabilidade	54
4.2.1	Interoperabilidade Horizontal	55
4.2.2	Interoperabilidade Vertical	55
4.3	Seleção de Casos de Estudo	56
4.3.1	Requisitos Comuns	56
4.3.2	Partilha do Registo Académico do Aluno	59
4.3.3	Agregação de Estatísticas sobre os Cursos	65
4.3.4	Concentração de Informação sobre os Recursos Humanos	67
4.3.5	Difusão de Notícias da Universidade	69
5	Arquitectura	73
5.1	Interoperabilidade Horizontal	74
5.2	Interoperabilidade Vertical	75
5.2.1	Agregação de Informação	76
5.2.2	Concentração de Informação	79
5.2.3	Difusão de Informação	80
6	Implementação	83
6.1	Opções Tecnológicas	84
6.1.1	XML Schema	84
6.1.2	Serviços Web	84
6.2	Convenções Técnicas	85
6.2.1	Idioma	85
6.2.2	Nomenclaturas	85
6.2.3	Codificação	86
6.2.4	Uso de Elementos ou Atributos	86
6.3	Partilha do Registo Académico do Aluno	86
6.3.1	Identificador do Registo Académico do Aluno	86
6.3.2	Representação do Registo Académico do Aluno	87
6.3.3	Especificação do Sistema	95
6.4	Agregação de Estatísticas sobre os Cursos	97
6.4.1	Identificador do Curso	97
6.4.2	Representação de Estatísticas sobre o Curso	98
6.4.3	Especificação do Sistema	101
6.5	Concentração de Informação sobre Recursos Humanos	102
6.5.1	Identificador do Recurso Humano	102
6.5.2	Representação de Informação sobre Recursos Humanos	102
6.5.3	Especificação do Sistema	103
6.6	Difusão de Notícias da Universidade	105
6.6.1	Identificador da Notícia	105
6.6.2	Representação de Notícias	106
6.6.3	Especificação do Sistema	108
6.7	Protótipos	110

7	Conclusões	111
7.1	Conclusões Gerais	112
7.2	Perspectivas de Trabalho Futuro	113
A	Estrutura da Base de Dados e Lista de Funcionalidades	123
A.1	Modelo Entidade-Relação	123
A.2	Lista de Funcionalidades	123
B	Especificação do Registo Académico do Aluno	131
B.1	Especificação do Elemento StudentRecord	131
C	Agregação de Estatísticas sobre Cursos	147
C.1	Especificação do Elemento DegreeStatistics	147
D	Concentração de Informação sobre Recursos Humanos	153
D.1	Especificação do Elemento StaffRecord	153
E	Difusão de Notícias da Universidade	161
E.1	Especificação do Elemento NewsItem	161

Lista de Figuras

2.1	As Hierarquias Paralelas na Burocracia Profissional	24
3.1	Arquitecturas Distribuídas: C/S, P2P e S/C	40
3.2	Sincronismo entre Sistemas Distribuídos: Síncrono e Assíncrono	41
3.3	Colocação dos Dados: Envio da Interrogação, Envio dos Dados e Solução Híbrida	43
3.4	Representação Lógica e Representação Física	44
3.5	Protocolos de Replicação de Dados	45
3.6	Propagação Síncrona e Propagação Assíncrona	46
3.7	Esquemas de Entrega dos Dados	49
3.8	Arquitetura EDI	49
3.9	Arquitetura baseada em Componentes	50
3.10	Arquitetura baseada em Serviços Web	52
4.1	Interoperabilidade Vertical e Horizontal	54
5.1	Agregação de Informação	76
5.2	Difusão de Informação - Instituições da Universidade e Público em Geral	82
6.1	Elemento StudentRecord	88
6.2	Elemento DocumentInfoType	88
6.3	Elemento StudentType	89
6.4	Elemento PersonType	90
6.5	Elementos AcademicDegreeType e DegreeType	92
6.6	Elementos AcademicSessionType e CourseType	94
6.7	Diagrama de Componentes para a Partilha de Informação sobre Alunos	95
6.8	Diagrama de Componentes para a Partilha de Informação sobre Alunos	97
6.9	Elemento DegreeStatisticsType	99
6.10	Elemento PersonsByEnrollmentsByCurricularPeriodType	100
6.11	Elemento PersonsByFinalGradeValueType	101
6.12	Diagrama de Componentes para a Agregação de Informação Estatística sobre Cursos	101
6.13	Diagrama de Sequência para a Agregação de Informação Estatística sobre Cursos	102
6.14	Elemento StaffRecord	103
6.15	Diagrama de Sequência para a Concentração de Informação sobre Recursos Humanos	105

6.16	Elemento <code>NewsItem</code>	107
6.17	Diagrama de Componentes para a Difusão de Notícias da Universidade .	108
6.18	Diagrama de Sequência para a Difusão Controlada de Notícias	109
6.19	Diagrama de Sequência para a Difusão Pública de Notícias	109
6.20	Arquitectura do protótipo relativo à representação de informação	110
A.1	Modelo Entidade-Relação	123

Lista de Tabelas

2.1	Sistemas de Informação Universitários avaliados	27
2.2	Número de funcionalidades por categoria	28
2.3	Número de funcionalidades por grupo e palavra-chave	29
2.4	Exemplo de funcionalidades mais comuns nos sistemas avaliados	29

Capítulo 1

Introdução

Neste capítulo, depois de um breve enquadramento, são referidas as principais motivações associadas à realização deste trabalho. No final, é apresentada a estrutura e organização do texto.

1.1 Enquadramento e Motivação

O crescimento explosivo da capacidade de processamento dos computadores e, mais recentemente, das redes de computadores, está na base de uma revolução nos sistemas de informação empresariais. Os sistemas centralizados (ou monolíticos) foram substituídos por sistemas de informação distribuídos, compostos por múltiplos computadores ligados em rede.

Numa primeira fase, o principal desafio foi a integração dos diversos sistemas informáticos existentes na empresa. Esta integração permitiu eliminar silos de informação e desenvolver sistemas que ofereciam uma visão global e integrada da organização. Mais recentemente, os avanços que estão na origem da Internet (tecnologia e infraestrutura) introduziram novos desafios às organizações. A ligação de sistemas tecnologicamente heterogéneos e geograficamente distantes permite olhar para além das fronteiras da própria organização. Actualmente, e um pouco por todas as indústrias, há uma forte aposta na integração dos sistemas a um nível inter-organizacional.

Na educação, os sistemas de informação universitários são, hoje, peças fundamentais das instituições académicas. Estão presentes em diversas áreas, desde a gestão financeira da organização até ao apoio na gestão curricular dos cursos. Nos últimos anos, tem-se assistido a um crescente investimento no desenvolvimento ou na aquisição deste tipo de sistemas.

No entanto, a colaboração entre estas instituições está ainda muito limitada ao nível da integração dos diferentes sistemas de informação. Por outro lado, a evolução da sociedade e das próprias organizações tem destacado a importância da interoperabilidade entre os vários sistemas. A título de exemplo, refere-se a declaração de Bolonha em que, dentre os principais objectivos, se estabelece a promoção da mobilidade de alunos, docentes e investigadores no espaço europeu.

As características actuais da Universidade do Porto, em termos de organização geográfica, torna ainda mais relevante a necessidade de integração. A Universidade do Porto está organizada em três grandes pólos, com algumas unidades situadas “extra-pólos”. O factor geográfico dificulta a partilha de informação entre as instituições e, em particular, o acesso por parte dos órgãos de gestão central.

Este trabalho vem ao encontro desta problemática, através do estudo e avaliação das alternativas para a interoperabilidade entre sistemas de informação universitários. Com esta dissertação, procura-se contribuir para a concepção e operacionalização de sistemas de informação distribuídos que sustentem as estratégias de integração inter-organizacional num contexto académico.

1.2 Organização da Dissertação

No Capítulo 2, é apresentado o conceito de Sistema de Informação Universitário. Depois de uma análise organizacional das universidades, é feita uma caracterização dos sistemas de informação existentes nas instituições de ensino superior. Para tal, foi construída uma base de dados das funcionalidades encontradas nestes sistemas. Conclui-se o capítulo, com a referência a algumas das principais iniciativas nesta área.

No Capítulo 3, aprofunda-se, com maior destaque para a vertente tecnológica, o conceito de interoperabilidade. É feita uma descrição dos níveis de interoperabilidade que é possível identificar, tendo por base duas classificações conhecidas, o LISI e o NMI. No resto do capítulo, procede-se a uma avaliação das opções existentes ao nível da arquitectura de sistemas distribuídos. No final, é feita uma breve revisão das principais alternativas tecnológicas.

No Capítulo 4, é apresentado o problema concreto de interoperabilidade entre sistemas de informação universitários, tendo como caso de estudo o cenário da Universidade do Porto. O problema foi estruturado segundo dois eixos: horizontal e vertical. Estes eixos são descritos e caracterizados neste capítulo. Na sequência desta caracterização, são identificados quatro padrões de interacção: partilha, agregação, concentração e difusão. Na conclusão do capítulo, e associados a cada um dos padrões, são seleccionados e apresentados quatro cenários de aplicação real.

No Capítulo 5, é feita uma apresentação detalhada da arquitectura proposta. Depois de descritas e justificadas as opções tomadas para cada um dos padrões de alto nível, é feita uma breve contextualização para cada caso de estudo particular.

No Capítulo 6, descreve-se a implementação dos casos de estudo anteriormente definidos. Após a apresentação das opções e convenções técnicas adoptadas, o capítulo está estruturado em função dos casos de estudo. No final, é feita uma breve referência aos protótipos implementados.

No Capítulo 7, são apresentadas as conclusões e algumas perspectivas de trabalho futuro.

No Anexo A, inclui-se a estrutura da base de dados construída para a compilação das funcionalidades encontradas nos sistemas de informação universitários, bem como a listagem completa destas funcionalidades.

Nos Anexos B, C, D e E incluem-se, respectivamente, as listagens das especificações para o registo académico do aluno, para as estatísticas sobre cursos, para as informações sobre recursos humanos e para as notícias.

Capítulo 2

Sistemas de Informação Universitários

Neste capítulo é apresentado o conceito de Sistema de Informação Universitário (SIU). Depois de uma breve abordagem às universidades como organizações, são apresentadas as principais características dos sistemas de informação existentes nas instituições de ensino superior. Aqui, expõe-se um estudo feito sobre as funcionalidades encontradas nestes sistemas. Na conclusão do capítulo é apresentado e descrito o problema da troca de informação entre sistemas deste tipo e são referidas algumas das principais iniciativas nesta área.

2.1 As Organizações Universitárias

As Universidades são organizações complexas. O leque de intervenientes que reúnem resulta num conjunto vasto e heterogéneo de interesses.

Mintzberg [70] classifica as universidades como organizações burocráticas mas não centralizadas, às quais dá o nome de Burocracias Profissionais. Associadas a este tipo de configuração, identificam-se características como a complexidade do trabalho desenvolvido pelos profissionais, a produção de serviços normalizados (pré-determinados ou previsíveis), uma administração ao serviço dos profissionais e uma estrutura paralela de apoio logístico e administrativo organizada segundo uma burocracia tradicional (mecanicista). A propósito destas duas hierarquias, ilustradas na Figura 2.1, Mintzberg afirma: “Na Burocracia Profissional, tem-se frequentemente duas hierarquias paralelas, uma para os profissionais, no sentido ascendente e que é de natureza democrática, e outra para as funções de apoio logístico, no sentido descendente e que tem a natureza de uma Burocracia Mecanicista.”.

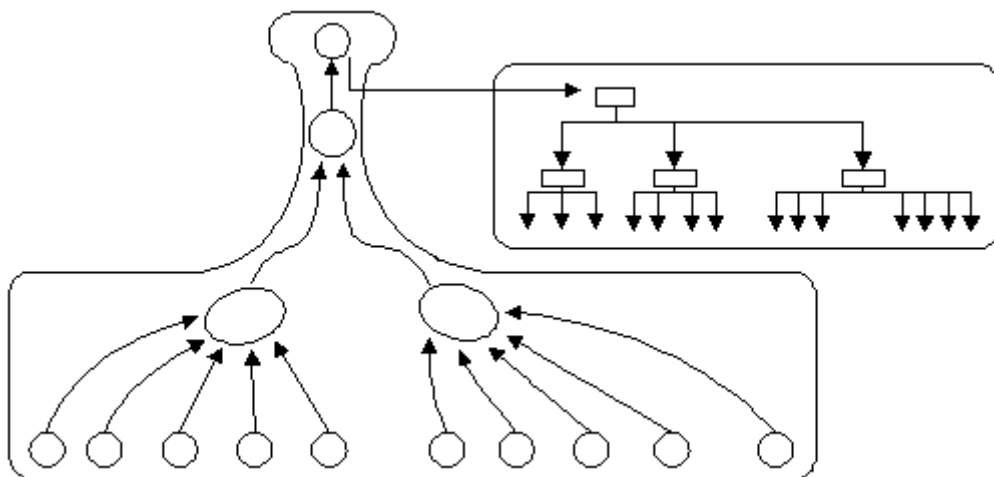


Figura 2.1: As Hierarquias Paralelas na Burocracia Profissional

A natureza complexa do trabalho executado e a necessidade de proximidade aos clientes (estudantes) fazem com que o profissional (docente) tenha, nestas organizações, grande liberdade de acção. O facto de grande parte do poder estar associado aos profissionais da instituição, resulta numa instituição muito descentralizada, onde coexistem uma grande diversidade de grupos e equipas com vasto grau de autonomia. Neste tipo de contexto, a formulação e aprovação de estratégias globais é tradicionalmente muito complexa. Este factor contribui para aumentar a inércia de organizações, tradicionalmente, pouco dinâmicas.

Os fluxos de informação que existem entre os profissionais de uma burocracia profissional são, assim, dispersos e pouco formais. A existência de vários subsistemas com hierarquias próprias, bem como o grande número de pontos de contacto com o sistema central, tem como consequência o aparecimento de fluxos de informação com padrões complexos e pouco estruturados. Na estrutura administrativa existente na organização

que, como já referimos, está organizada segundo uma burocracia centralizada, com procedimentos formalizados, o fluxo de informação está mais bem definido.

Deste modo, é possível identificar duas estruturas dentro de uma mesma organização com atitudes muito diferentes perante a gestão da informação. Por um lado, a estrutura composta pelos profissionais apresenta um padrão muito descentralizado e pouco definido. Por outro, a estrutura administrativa de apoio é centralizada e formalizada. Assim, a definição de estratégias para a gestão de informação será, necessariamente, muito diferente num cenário e no outro.

2.2 Os Sistemas de Informação nas Universidades

Podemos dividir a aplicação das tecnologias de informação numa instituição de ensino superior em dois grupos: tecnologias de apoio e coordenação da actividade dos membros da instituição e tecnologias de apoio directo ao ensino. Nesta dissertação, procura-se aprofundar o estudo dos sistemas de informação associados ao primeiro grupo, geralmente designados por Sistemas de Informação Universitários.

O conceito de Sistemas de Informação Universitários (SIU) pode considerar-se próximo da definição base de Sistema de Informação (SI), ou seja, “um conjunto de componentes interrelacionados que reúnem, processam, armazenam e distribuem informação para suportar a tomada de decisão, coordenação e controlo numa organização” [63]. Sistemas de Informação Universitários correspondem à aplicação de SI ao contexto universitário, o que se traduz em realçar as dimensões de arquivo e de comunicação de um repositório comum de informação. Na realidade, um dos papéis mais importantes desempenhados pelos SIU é o de arquivo e registo da actividade universitária.

Sobre os sistemas técnicos existentes nas burocracias profissionais, e como resultado da grande autonomia dos profissionais, Mintzberg refere que “não são nem muito sofisticados, nem muito automatizados, nem muito reguladores” e que “a tecnologia da organização - os conhecimentos que utiliza - é sofisticada; mas o seu sistema técnico - o conjunto dos instrumentos que utiliza - não o é.” [70].

A evolução dos sistemas de informação dentro destas organizações pode ser analisada segundo duas perspectivas. Por um lado, a estrutura administrativa permitiu a existência de uma visão global e, conseqüentemente, o desenvolvimento de sistemas de informação de forma coordenada. Por outro lado, a introdução de sistemas de informação junto da estrutura profissional das universidades foi descoordenada. A natureza descentralizada desta estrutura resultou num desenvolvimento disperso pelos departamentos, núcleos e equipas, de nível muito desigual de sector para sector, com ilhas de organização num contexto geral de impossibilidade prática de agregação de informação. Assim, se, por um lado, houve melhorias no tratamento de informação, por outro, aumentou-se o esforço associado ao tratamento de dados duplicados e aos vários pontos de acesso ao sistema, com informação parcelar, incoerente e não integrada.

Algumas iniciativas no sentido de centralizar estes sistemas constituíram o passo seguinte. Em muitos casos, o desenvolvimento interno foi adoptado como método principal de desenvolvimento, em particular nos módulos de gestão de alunos [66]. A existência de competências internas, os factores relacionados com a oportunidade de investigação e a falta de alternativas comerciais viáveis, justificaram esta opção [60].

A implementação de Portais Universitários tem sido uma tendência generalizada no meio académico [32]. Para além de atingir o objectivo de satisfazer as necessidades dos estudantes (em particular a mobilidade), Eisler enumera um conjunto de razões para justificar esta tendência: a necessidade de permitir o acesso aos dados e aos processos de trabalho por parte do pessoal da instituição, o aumento da concorrência para a captação de estudantes, o desejo de ligar os antigos estudantes à instituição e a importância de promover a instituição junto do público [32]. Estas soluções de tipo portal surgem não por opção, mas pela necessidade de agregar informação muito dispersa e reunir num único ponto um conjunto de ligações para os vários silos de informação existentes na instituição.

Em muitos casos estas soluções resultam na conjugação, numa mesma página, de um conjunto de recursos só levemente relacionados e que de facto não correspondem a um sistema de informação integrado.

Como resultado do aumento do peso organizacional das unidades dedicadas ao desenvolvimento e manutenção dos SIU, constituíram-se associações especificamente dedicadas a estes assuntos como é o caso da EUNIS e da EDUCAUSE.

A European University Information Systems Organization (EUNIS) é uma associação fundada em 1993 por representantes de 8 países, actualmente (Setembro de 2003) são 22, que tem como principal objectivo a troca de experiências entre responsáveis pelos sistemas de informação das instituições universitárias na Europa. Desde 1995 organiza uma conferência anual [35].

A EDUCAUSE é a congénere nos EUA da EUNIS. É uma organização sem fins lucrativos que reúne mais de 1800 escolas, e tem como missão a promoção das tecnologias de informação no ensino superior [30]. Organiza, desde 1993, uma conferência anual.

2.3 Funcionalidades dos SIU

Há um grande número de funcionalidades que se enquadram no âmbito do que designamos por SIU e que resultam num conjunto de sistemas muito diversificados. Assim, no sentido de caracterizar e aprofundar o conhecimento sobre este tipo de sistemas, foi feito um estudo sobre as funcionalidades mais comuns. É necessário referir que esta avaliação se cingiu a alguns exemplos considerados significativos e não pretende representar a globalidade dos sistemas existentes.

Foram estudados 15 SIU diferentes, onde se incluem sistemas desenvolvidos pelas próprias instituições, sistemas comerciais e um sistema desenvolvido em regime de código livre (*open-source*), utilizado em mais de 100 instituições norte americanas [74]. Na Tabela 2.1, procede-se à enumeração dos sistemas avaliados.

Tipo	Instituição	URL / Referência
Desenvolvido pela Instituição		
	Technical University of Denmark Graz University of Technology, Austria Kaunas University of Technology, Lithuania University of Kent at Canterbury, United Kingdom University of Ljubljana, Faculty of Computer and Information Science, Slovenia University of Paisley, Scotland - United Kingdom Faculdade de Engenharia da UP, Portugal Slovak Agricultural University in Nitra, Slovak Republic University of St. Andrews, Scotland Instituto Politécnico de Tomar, Portugal Faculty of Mathematics, Informatics, and Mechanics, Warsaw University, Poland	http://www.dtu.dk http://www.tugraz.at [53] http://www.ktu.lt [78] http://www.ukc.ac.uk [46] http://www.fri.uni-lj.si [85] http://www.paisley.ac.uk [92] http://sifeup.fe.up.pt http://www.uniag.sk [100] http://www.st-and.ac.uk [64] http://www.ipt.pt [98] http://www.mimuw.edu.pl [69]
Comercial		
	Campus Pipeline, Inc Digitalis Informática, Lda Sophia - Universidade Católica Portuguesa (GIPSI)	http://www.campuspipeline.com http://www.digitalis.pt http://www.gipsi.ucp.pt
Código Livre		
	uPortal	http://mis105.mis.udel.edu/jasig/uportal/

Tabela 2.1: Sistemas de Informação Universitários avaliados

O método de estudo a que se recorreu na análise de sistemas variou significativamente de caso para caso. Houve situações em que foi possível usar o sistema directamente, noutros casos foram consultados artigos disponíveis sobre o sistema. Os artigos publicados na conferência EUNIS 2002: “The Changing Universities: The Challenge of New Technologies” revelaram-se um recurso importante [82]. Como consequência, os níveis de detalhe obtidos variam muito de sistema para sistema. Em alguns casos, a informação disponível resume-se a um pequeno conjunto de funcionalidades relacionadas com determinada área do sistema. Noutros foi possível fazer uma avaliação completa das funcionalidades.

A fim de organizar a informação reunida, foi construída uma base de dados, com informação sobre os sistemas, as funcionalidades, as categorias e palavras-chave. No Apêndice A encontra-se a especificação da estrutura da base de dados e a listagem com-

pleta das funcionalidades encontradas.

As funcionalidades foram organizadas, de forma não exclusiva, segundo categorias, definidas durante o decurso do trabalho. Cada funcionalidade pode estar associada a mais do que uma categoria. Na Tabela 2.2 apresenta-se o número de funcionalidades por categoria.

Categoria	Número de Funcionalidades
Administrativa	99
Académica	78
Sistema	23
Investigação e Desenvolvimento	16

Tabela 2.2: Número de funcionalidades por categoria

Na categoria Administrativa reúnem-se as funcionalidades associadas a tarefas administrativas: gestão de um calendário do curso, gestão das classificações dos alunos e gestão do registo académico do aluno.

A categoria Académica agrega as funcionalidades relacionadas com a actividade lectiva; reúne funcionalidades como a gestão de sumários das aulas, disponibilização do catálogo dos cursos e gestão de horários.

A categoria Sistema corresponde a funcionalidades associadas à administração ou configuração do próprio sistema de informação. Por exemplo, a criação de novos utilizadores no sistema.

Por fim, na categoria Investigação e Desenvolvimento incluem-se as funcionalidades relacionadas com as actividades de investigação: a pesquisa de publicações, o registo das actividades de investigação e a gestão de informação relativa a projectos.

Para além da definição de categorias, também foram definidas palavras-chave, com o intuito de permitir uma forma de caracterização adicional mais flexível para cada funcionalidade. Cada funcionalidade pode ter associadas várias palavras-chave. Depois de estruturadas as palavras-chave, definiram-se grupos com base no trabalho de Coimbra e Rito Silva [11] na classificação de processos académicos no contexto do Ensino Superior. Foram definidos 4 grupos: actividade, suporte, indivíduo e organização. Na Tabela 2.3 é discriminado o número de funcionalidades associadas a cada palavra-chave.

Grupo	Palavra-Chave	NºFuncionalidades
Suporte	Ferramenta	125
Suporte	Administrativo	104
Indivíduo	Aluno	91
Indivíduo	Pessoal Docente	69
Actividade	Disciplina	56
Suporte	Colaboração	49
Indivíduo	Investigador	32
Organização	Instituição	28
Suporte	Sistema de Informação	24
Actividade	Pós-Graduação	18
Actividade	Programa	18
Indivíduo	Pessoa	18
Actividade	Grau Académico	16
Actividade	I&D	14
Organização	Grupo / Unidade	13
Organização	Departamento	13
Suporte	Financeiro	11
Indivíduo	Pessoal	11
Indivíduo	Antigos Alunos	2

Tabela 2.3: Número de funcionalidades por grupo e palavra-chave

Com base nos dados apresentados e no estudo feito, é possível afirmar que, nos sistemas escolhidos, se destacam as funcionalidades associadas ao suporte, juntamente com as palavras-chave associadas a ferramentas, tarefas administrativas e alunos. De facto, as funcionalidades mais comuns são aquelas relacionadas com a gestão administrativa dos indivíduos (alunos, docentes e pessoal). São, também, frequentes as funcionalidades associadas à gestão das disciplinas e dos cursos. Por sua vez, as ferramentas também são frequentes: sistema de *help-desk*, pesquisa de pessoal, gestão de grupos de discussão.

Na Tabela 2.4 apresentam-se as funcionalidades mais comuns com indicação do número de sistemas onde foram encontradas.

Gestão das Matrículas nas Disciplinas	7
Ficha de Curso	5
Ficha da Disciplina	5
Ficha do Aluno	5
Ficha do Docente	4

Tabela 2.4: Exemplo de funcionalidades mais comuns nos sistemas avaliados

No outro extremo, temos funcionalidades muito específicas que encontramos, pontualmente, em alguns sistemas, como:

- Integração com Sistema de Leitura Óptica.

- Gestor de Bookmarks Personalizado.
- Gestor de Galerias de Fotografias.

Tendo em conta o tema central desta tese, no âmbito da interoperabilidade foram registadas apenas 3 funcionalidades:

- Exportação da lista de alunos por disciplina.
- Exportação de dados para a impressão de cartões de alunos.
- Exportação dos horários (p.e. para Excel).

No âmbito dos SIU, os trabalhos nesta área são ainda raros, pelo que é natural a reduzida lista de funcionalidades catalogadas neste domínio.

Concluindo, estes resultados vêm ao encontro do esperado em organizações com esta configuração. Os procedimentos administrativos de apoio às actividades primárias são os que se encontram mais normalizados, sendo por isso mais fáceis de transpor para o âmbito de um sistema de informação. Por outro lado, os processos associados ao ensino ou investigação são mais subjectivos e, consequentemente, mais difíceis de representar.

2.4 Troca de Informação

2.4.1 Definição

Os sistemas de informação têm fronteiras que delimitam o seu âmbito. No caso dos sistemas de informação universitários, o contexto abrangido é vasto e resulta de uma organização complexa e fortemente descentralizada.

No entanto, tal como a própria organização, os sistemas de informação necessitam de interagir com outros sistemas externos. É essencial a possibilidade de troca de registos de alunos entre instituições, a consulta de informação institucional, disponibilização de currículos ou estatísticas de desempenho. A estes processos de partilha e reutilização de informação e procedimentos entre sistemas, está associado o termo “interoperabilidade”. Beynon-Davies define interoperabilidade como “uma medida do grau segundo o qual os sistemas de informação são capazes de se coordenar e colaborar” [6].

O grau de interoperabilidade entre dois sistemas pode ser avaliado segundo vários níveis. Miller propõe uma divisão em 6 dimensões [68]:

- Interoperabilidade Técnica: relacionada com normas relativas a comunicação, transporte, armazenamento e representação.
- Interoperabilidade Semântica: relacionada com a construção semântica, meta-dados e estrutura.
- Interoperabilidade Política / Humana: relacionada com a dimensão organizacional, os fluxos de trabalho e de poder e as culturas.

- Interoperabilidade Inter-Comunitária: relacionada com a interacção entre domínios / contextos independentes (p.e.: a troca de informação entre uma instituição bancária e uma faculdade).
- Interoperabilidade Legal: relacionada com decisões legais, protecção de dados.
- Interoperabilidade Internacional: relacionada com a nacionalidade e com impacto nas anteriores dimensões. (p.e.: a troca de informação sobre alunos do programa ERASMUS).

A declaração de Bolonha [36], assinada em 1999 pelos ministros da educação dos países da UE, veio reafirmar a importância estratégica de uma aposta na interoperabilidade entre os sistemas de informação das instituições de educação europeias. Dos objectivos traçados, destacam-se a aposta na promoção da mobilidade de alunos, docentes, investigadores e outro pessoal, e a aposta na cooperação entre as instituições.

As instituições europeias põem como condição para a participação e candidatura a programas de mobilidade, a existência de informação em formato electrónico armazenada em sistemas de informação.

2.4.2 Iniciativas Existentes no Contexto Académico

São várias as iniciativas a nível mundial que exploram a colaboração entre instituições do ensino superior [2], embora a maioria dos projectos existentes incida sobretudo na troca de recursos educacionais (instrução), o que está fora do âmbito desta dissertação. Nesta secção, faz-se uma breve apresentação de algumas das principais iniciativas existentes.

UN/CEFACT

A sigla UN/CEFACT corresponde às iniciais de United Nations Centre for Trade Facilitation and Electronic Business. Este centro das Nações Unidas tem como missão promover mecanismos que simplifiquem as transacções comerciais, de bens ou serviços, entre empresas, e, desta forma, contribuir para o crescimento global do comércio. Sob a alçada desta organização, coexistem várias iniciativas: o caso do grupo de trabalho UN/EDIFACT e das especificações ebXML.

O grupo de trabalho UN/EDIFACT é responsável pelo desenvolvimento, manutenção e promoção da norma EDI - UN/EDIFACT. Em 1999, o UN/CEFACT lançou uma iniciativa com vista ao desenvolvimento de estruturas em XML para a promoção do comércio global, mais tarde designada por ebXML [26].

No âmbito do UN/CEFACT, tem sido dada prioridade ao desenvolvimento de normas para a indústria e negócios. Existe pouco material publicado, relevante para a área da educação. Apesar de tudo, esta organização, em parceria com o grupo de normas OASIS, tem-se afirmado como um dos principais intervenientes no mercado das normas para a transferência electrónica de informação. O facto de estar sob a alçada da ONU garante-lhe uma visibilidade acrescida.

ANSI ASC X12A

O ACS X12 é um comité, certificado pela ANSI em 1979, que visa o desenvolvimento de normas para a implementação de transacções electrónicas entre empresas de uma mesma indústria. As normas definidas pelo ACS X12 são utilizadas para a definição da sintaxe, estrutura e ordenação das mensagens utilizadas nas transacções electrónicas entre empresas e são designadas por Electronic Data Interchange (EDI).

Actualmente, o trabalho desenvolvido pelo ACS X12 no âmbito da educação é mais abrangente do que aquele que tem vindo a ser realizado pelo CEFAC. O subcomité X12A [103] tem como âmbito de trabalho a administração escolar. As iniciativas levadas a cabo pelo X12A compreendem todos os níveis de escolaridade, desde o primário ao universitário, bem como casos especiais de ensino (privado, profissional ou outro). As normas já desenvolvidas abrangem a área administrativa, o registo académico dos alunos, informações relativa aos recursos humanos, a ajuda financeira e a gestão curricular. São os formatos mais usados nos EUA para a transferência de informação, maioritariamente administrativa, entre instituições académicas. Os formatos definidos incluem: TS130 - Student Educational Record, TS133 - Educational Institution Record, TS152 - Statistical Government Information, entre outros.

Ambas as normas, X12 e EDIFACT, permitem codificar os documentos (estruturados em blocos de dados delimitados) em formatos que podem ser interpretados por aplicações informáticas. A transmissão dos dados não é especificada nas normas, sendo frequente o recurso a VAN (Value Added Networks).

SPEEDE/ExPRESS

SPEEDE/ExPRESS [71] são as iniciais de Standardization of Postsecondary Education Electronic Data Exchange / Exchange of Permanent Records Electronically for Students and Schools. Este projecto, iniciado em 1988 e patrocinado pelo National Center for Education Statistics dos EUA, constitui uma implementação de um conjunto de formatos para o uso de EDI na educação. Estas normas são mantidas pelo ASC X12 [102] (ver secção anterior).

Postsecondary Electronic Standards Council

A Postsecondary Electronic Standards Council (PESC) [18] é uma organização dos EUA, constituída por cerca de 50 membros (instituições académicas e empresas comerciais), e que visa a promoção do uso de normas electrónicas na educação e a respectiva implementação. Destacam-se as actividades desenvolvidas no âmbito do grupo XML Forum [21], que se posiciona como o grupo responsável pelas normas XML na área da educação [20]. Um dos principais objectivos do PESC é o apoio na transição de tecnologias e mecanismos baseados em EDI para XML.

Nos documentos já publicados por este grupo, inclui-se a Especificação Técnica em XML para o Ensino Superior [22] e um conjunto de livros brancos sobre uso de chaves públicas, identificadores de estudantes e XML.

Algumas das normas já existentes e aplicadas no meio académico, em particular as que são definidas pelo ACS X12, estão a ser usadas como ponto de partida para o trabalho desenvolvido pelo XML Forum. No conjunto de tarefas agendadas, está incluída

a conversão para XML de algumas das especificações em X12 [17]. Recentemente, foi publicada a versão 1.0 do XML Postsecondary Transcript Schema [19].

Sistema de Codificación Académica Normalizado en Red

O Sistema de Codificación Académica Normalizado en Red (SCANet) [89] é um projecto iniciado em 2001, na Universidade Espanhola de Lérida, que visa o desenvolvimento de normas para a representação e transferência de informação no contexto da gestão académica. O projecto tem como objectivo “tornar mais fáceis as relações entre as universidades, destas com os estudantes e docentes, trazendo valor acrescentado aos processos de gestão académica” [91].

O projecto está estruturado em 3 fases [90]:

- Definição de códigos de identificação únicos.
- Especificação de requisitos para transferência de conteúdos electrónicos.
- Harmonização das formas e procedimentos.

Destaque para as normas já publicadas:

- Norma 1.01/02E - Identificación Normalizada del Código Unificado de Estudiante [86].
- Norma 2.01/02E - Código de Universidad y Códigos de servicios asociados [87].
- Norma 4.01/03E - Homologación de Datos de Expedientes de Primer y Segundo Ciclo para Intercambios, Traslados y Certificaciones Académicas [88].

Actualmente o projecto conta com a participação de cerca de 50 universidades espanholas e com a colaboração do Governo Central Espanhol na elaboração de regulamentos administrativos [55].

Internet2 Middleware Initiative

A Internet2 Middleware Initiative (I2-MI) [59] é um projecto enquadrado na Iniciativa Norte Americana Internet2, que visa a promoção e desenvolvimento de serviços e infra-estruturas que permitam a integração entre aplicações no contexto universitário e comunidades relacionadas. Esta iniciativa abrange áreas como segurança, partilha de informação, autenticação e uso de chaves públicas.

Dentre os vários projectos e iniciativas desenvolvidas no contexto da I2-MI, destacam-se as seguintes:

- Middleware Architecture Committee for Education (MACE) - grupo, constituído por universidades e que está envolvido em actividades nas áreas de directórios, infra-estrutura de chaves públicas (PKI), autorização e autenticação.

- No âmbito do MACE destacam-se os projectos eduPerson, eduOrg [31] e DoDHE [57]. Os dois primeiros têm como objectivo a definição de objectos LDAP para a representação de indivíduos e organizações no contexto académico. O projecto Directory of Directories for Higher Education (DoDHE) insere-se no estudo de tecnologias que apoiem a integração inter-institucional de pesquisas em directórios.
- O projecto Shibboleth [58] tem por objectivo o estudo e desenvolvimento de arquitecturas, políticas, tecnologias e implementações que apoiem a partilha de recursos Web, sujeitos a controlo de acessos. Este projecto envolve, ainda, o desenvolvimento de uma estrutura para a interoperabilidade dentro da comunidade do ensino superior.
- Multicampus Middleware é um grupo constituído em 2002 e que tem como objectivo fomentar a troca de experiências no âmbito da implementação de tecnologias de middleware em ambientes universitários.

Schools Interoperability Framework

O Schools Interoperability Framework (SIF) [93] é um consórcio empresarial dos EUA que, desde 1997, procura desenvolver uma especificação para a integração de aplicações de âmbito educacional. A especificação publicada permite, por exemplo, a interoperabilidade entre aplicações relacionadas com a gestão de matrículas de alunos, a gestão da cantina ou a emissão de cartões. A especificação desenvolvida baseia-se na norma XML do W3C e a versão mais recente disponível é a 1.1. Este trabalho está direccionado para o ensino pré-universitário e tem uma forte adesão por parte dos principais fabricantes de aplicações.

A arquitectura proposta pelo SIF é baseada num modelo distribuído, em que as aplicações informáticas recorrem a agentes para comunicar com um nó intermediário central - o Zone Integration Server (ZIS). Esta configuração existe em cada instituição de ensino.

A nível europeu, o projecto OASIS - Open Architecture and Schools in Society [73], enquadrado no programa de apoio às Tecnologias de Informação na Sociedade (IST), procura adaptar o trabalho desenvolvido pelo SIF ao contexto europeu [10].

IEEE Learning Technology Standards Committee

O IEEE Learning Technology Standards Committee (LTSC) [95] é a entidade, enquadrada no contexto da IEEE Computer Society Standards, responsável pelo desenvolvimento de normas técnicas, recomendações e linhas de acção no contexto das tecnologias de ensino focadas sobretudo na instrução.

Instructional Management Systems

O Instructional Management Systems (IMS) [56] é uma das principais iniciativas no âmbito do ensino online. É uma organização sem fins lucrativos constituída nos EUA em 1999, com mais de 100 membros activos, entre instituições académicas e empresas.

Tem publicado um número significativo de especificações, baseadas em XML, para a troca de conteúdo educacional e informação sobre os educadores, entre sistemas educacionais.

DCMI Education Working Group

O DCMI-EWG tem por objectivo o desenvolvimento de elementos no âmbito da DCMI, tendo em vista a caracterização de materiais educacionais. A Dublin Core Metadata Initiative (DCMI) constituiu este grupo de trabalho na área da educação, em 1999 [27].

Capítulo 3

Tecnologias de Apoio à Interoperabilidade

Neste capítulo aprofunda-se, com maior destaque para a vertente tecnológica, o conceito de interoperabilidade. É feita uma descrição dos níveis de interoperabilidade que é possível identificar, tendo por base duas classificações conhecidas, o LISI e o NMI (Secção 3.1). Nas secções seguintes, procede-se a uma avaliação das opções existentes ao nível da arquitectura de sistemas distribuídos (Secção 3.2 e 3.7). Nesta avaliação, destacam-se as dimensões consideradas mais relevantes no contexto da interoperabilidade entre SIU. No final, é feita uma breve revisão das principais alternativas tecnológicas (Secção 3.8).

3.1 Níveis de Interoperabilidade

Na Secção 2.4.1, foi apresentada uma definição para Interoperabilidade no âmbito dos Sistemas de Informação e segundo seis dimensões. Neste capítulo, procede-se a uma análise mais detalhada das dimensões técnica e semântica.

Existem vários modelos de referência que procuram caracterizar os níveis de interoperabilidade técnica possíveis entre dois sistemas. A comunidade militar tem desenvolvido muito trabalho nesta área e os dois principais modelos publicados surgem neste contexto - o *Level of Information Systems Interoperability* (LISI) e o *NATO C3 Technical Architecture (NC3TA) Reference Model for Interoperability* (NMI) [99].

O modelo LISI foi publicado pelo Departamento de Defesa dos EUA [49] e identifica quatro domínios: procedimentos e políticas, aplicações, dados e infraestrutura. Em cada um destes domínios, são enumerados níveis de interoperabilidade, que dão origem a cinco categorias de interoperabilidade técnica:

- **Nível 0:** Isolado (Manual) - Não ligado, mecanismos manuais (p.e. disquete).
- **Nível 1:** Ligado (P2P) - Ligação electrónica; dados e aplicações separados (p.e. correio electrónico).
- **Nível 2:** Funcional (Distribuído) - Funções mínimas comuns; dados e aplicações separados (p.e. HTTP).
- **Nível 3:** Domínio (Integrado) - Dados partilhados; aplicações separadas.
- **Nível 4:** Empresa (Universal) - Manipulação interactiva; dados e aplicações partilhados.

O NMI está incluído na *NATO Consultation, Command and Control (C3) Technical Architecture* (NC3TA) [50] e estabelece graus e subgraus de interoperabilidade. Os graus de interoperabilidade definem um modelo de maturidade que captura a sofisticação da interoperabilidade; por sua vez, os subgraus descrevem um modelo de capacidades que reflectem a funcionalidade disponível.

- **Grau 1:** Troca de Dados Não Estruturados.
Este nível envolve a troca de dados não estruturados, interpretáveis por humanos (p.e.: papéis, relatórios, etc).
- **Grau 2:** Troca de Dados Estruturados.
Este nível envolve a troca de informação interpretável por humanos, destinada a processamento manual ou automático. No entanto, são necessários mecanismos manuais de compilação, recepção e/ou envio.
- **Grau 3:** Partilha de Dados.
Este nível envolve a troca automática de dados sobre sistemas com modelos de partilha comuns.
- **Grau 4:** Partilha de Informação.
É uma extensão do grau 3 e estabelece a interpretação universal de informação através do processamento cooperativo dos dados.

A não conectividade entre sistemas (Nível 0 do LISI) não é mencionada neste modelo. Nos restantes níveis, existe uma correspondência entre os dois modelos.

Estes modelos definem uma estrutura genérica para a avaliação do grau de interoperabilidade entre dois sistemas. É possível sistematizar a avaliação de arquitecturas deste tipo, discriminando três camadas [67]:

- **Comunicação**

Esta camada inclui os protocolos de comunicação para a troca de mensagens entre as partes. Podem ser usados protocolos privados ou públicos, como o HTTP. Permite estabelecer um acordo ao nível da infraestrutura tecnológica.

- **Conteúdo**

Nesta camada, enquadram-se as linguagens e os modelos que permitem descrever e organizar a informação, de forma a ser entendida pelas partes. Aborda não só aspectos técnicos mas, também, questões semânticas.

- **Processo**

Esta camada relaciona-se com as interacções entre os processos de negócio, com os significados de determinada acção, com o tipo de acções que são permitidas, com o teor das respostas esperadas, etc. Estabelece uma semântica comum.

Em suma, Interoperabilidade Técnica e Semântica é a capacidade que um sistema tem para aceder a partes, dados ou funcionalidades, de outro sistema. Ao nível mais simples, situa-se a troca de dados entre dois sistemas de forma manual e com intervenção humana (rotinas de exportação e importação de dados entre aplicações desligadas). O grau máximo de interoperabilidade é atingido quando os dois sistemas partilham os objectos de dados e os serviços, de forma harmoniosa e invisível para o exterior, sem intervenção humana.

3.2 Arquitecturas Distribuídas

Uma das dimensões que se pode considerar para classificar os sistemas distribuídos tem a ver com os papéis desempenhados pelos *Servidores* e *Clientes*. A designação *Servidor* é atribuída ao sistema que disponibiliza os dados ou o serviço. O *Cliente* é o sistema que procura os dados ou o serviço. Tendo esta classificação presente, podemos distinguir 3 arquitecturas para os sistemas distribuídos (Figura 3.1):

- **Cliente-Servidor:** A interacção é iniciada pelo cliente. São geralmente designadas por arquitecturas do tipo *Pull*. Uma vantagem desta arquitectura é a elevada qualidade da informação que circula, uma vez que apenas é enviada aquela que é explicitamente pedida pelos clientes. É uma arquitectura vantajosa em cenários de entrega de informação já existente (por oposto a informação alterada ou nova). Num sistema com um número reduzido de servidores e um número muito elevado de clientes, esta arquitectura pode ser inadequada devido à sobrecarga dos servidores. Um exemplo de uma arquitectura deste tipo é a navegação na WWW.

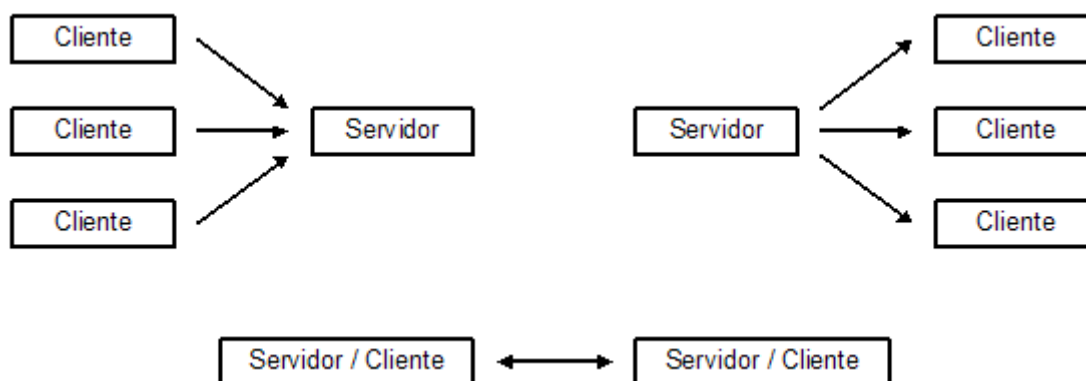


Figura 3.1: Arquitecturas Distribuídas: C/S, P2P e S/C
Arquitecturas Distribuídas: C/S, P2P e S/C (Adaptado de [7])

- **Peer-to-peer:** Os diversos sistemas tanto podem desempenhar funções de cliente, como de servidor. A interação pode ser iniciada por qualquer uma das partes. As principais vantagens desta arquitectura é a grande tolerância a falhas e escalabilidade. No entanto, são sistemas de implementação complexa uma vez que não existe controlo sobre o papel desempenhado por cada sistema (cliente ou servidor) e sobre a localização dos dados. Apesar de um custo inicial baixo, a administração e manutenção tendem a ser mais complexas a médio prazo [61]. A segurança é um dos aspectos mais frágeis. Um exemplo de um sistema com esta arquitectura é o Napster.
- **Servidor-Cliente:** Neste cenário, o servidor inicia a comunicação. São geralmente designadas por arquitecturas do tipo *Push*. É uma arquitectura adequada a cenários de envio de itens novos ou alterados. É também vantajosa em cenários de difusão de informação. Por outro lado, um sistema deste tipo pode não ser interessante quando os clientes têm pedidos irregulares, porque os clientes são interrompidos várias vezes para processar informação que não é do interesse deles. Um exemplo de uma arquitectura deste tipo são os grupos de discussão baseados em correio electrónico.

3.3 Modelos de Programação

Existem diversas abordagens à categorização dos vários modelos de programação distribuída ([96], [34], [83]). Refere-se aqui a classificação definida por Emmerich [33]:

- **Procedimental:** As *Remote Procedure Calls* (RPC) foram concebidas pela Sun Microsystems no início dos anos 80. O conceito base consiste em permitir a invocação remota de procedimentos publicados por servidores, da mesma forma que são feitas invocações locais.
- **Transaccional:** Permite gerir transacções envolvendo múltiplos componentes em diferentes servidores. Em geral, usam protocolos do tipo “*two-phase commit*” na implementação. Exemplos: IBM CICS e BEA Tuxedo.

- **Orientado às Mensagens:** A comunicação entre sistemas distribuídos é implementada recorrendo à troca de mensagens. O IBM MQSeries e o Sun JMQ são exemplos de produtos desta categoria.
- **Baseado em Objectos e Componentes:** Constituem uma evolução dos tradicionais RPCs, de forma semelhante à evolução que se assistiu nas linguagens de programação. A ideia é permitir a colaboração entre sistemas ao nível de componentes/objectos, de forma similar à que existe entre procedimentos no RPC. Exemplos: CORBA, Microsoft COM e Sun RMI.

3.4 Sincronismo

Uma outra dimensão que pode ser usada para classificar sistemas distribuídos está relacionada com o sincronismo ou acoplamento entre os sistemas. Há autores que estendem esta classificação e consideram três subníveis de acoplamento extra: temporal, espacial e de fluxo [34]. Nesta secção, descrevem-se os dois tipos de sincronismo de um ponto de vista genérico (Figura 3.2).

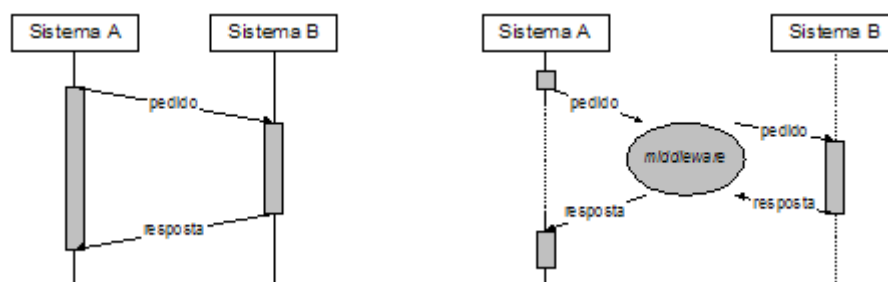


Figura 3.2: Sincronismo entre Sistemas Distribuídos: Síncrono e Assíncrono

3.4.1 Interação Síncrona

Nas arquitecturas cliente-sevidor tradicionais, este tem sido o modelo predominante [47]. Neste cenário, os sistemas que participam na comunicação necessitam de estar activos e ocupados em simultâneo, no decorrer da transacção.

O modelo de programação baseado em RPC é um dos principais exemplos de uma arquitectura síncrona. Estas soluções procuram esconder a “distribuição” dos intervenientes, tornando as transacções iguais a invocações locais de procedimentos.

Os sistemas síncronos dão origem a um arquitectura pouco dinâmica, mais adequada a ambientes muito controlados. São soluções que se adaptam mal a sistemas de grande e variável dimensão, envolvendo várias organizações, devido aos custos de coordenação (mensagens e processamento). Por outro lado, como vantagem, a implementação é menos complexa do que a alternativa assíncrona.

3.4.2 Interacção Assíncrona

A crescente necessidade de flexibilidade, fiabilidade e desacoplamento em aplicações distribuídas de grande escala, em particular na Internet, resultou numa tendência generalizada para a adopção de protocolos assíncronos [47]. As arquitecturas baseadas na passagem de mensagens são um exemplo de uma solução assíncrona. Nestes casos, o cliente comunica com o servidor enviando mensagens estruturadas através de interfaces simples e normalizadas. Toda a informação sobre o método a invocar e dados a transmitir está incluída na mensagem. A lógica da interacção é deslocada para o nível da mensagem, que pode conter dados, pedidos de acção ou respostas.

É uma abordagem que constitui um compromisso entre o nível de abstracção programático (menor do que o modelo anterior) e os requisitos práticos (aplicações fortemente desacopladas) [47]. O EDI (ver Secção 3.8.1) é um exemplo de uma arquitectura assíncrona baseada na troca de mensagens. A primeira vaga de Serviços Web foi desenhada como RPC, mas, devido às vantagens referidas, a tendência actual é a adopção de mecanismos baseados na passagem de documentos XML [8].

As abordagens assíncronas são, do ponto de vista da implementação, soluções mais complexas. Requerem a existência de mecanismos intermédios para a gestão da interacção - gestão de filas de espera, gestão dos sistemas participantes, endereçamento das invocações, etc.

3.5 Colocação dos Dados

No contexto do processamento da interrogação em ambientes distribuídos, existem duas áreas onde é possível introduzir optimizações: na *execução da interrogação* e na *colocação dinâmica dos dados* [61]. Com a colocação dinâmica dos dados, procura-se optimizar a colocação dos dados em função dos locais onde são necessários com maior frequência, usando mecanismos de *caches* ou *replicação* de informação.

Existem três abordagens alternativas, associadas à colocação dos dados ([61], [62] e [44]): mover a interrogação para os dados (execução no servidor, *query shipping*), mover os dados para a interrogação (execução no cliente, *data shipping*) ou uma solução híbrida (*hybrid shipping*) (Figura 3.3).

3.5.1 Mover a Interrogação

Nesta alternativa, a execução das interrogações é sempre feita no servidor onde estão os dados. O cliente envia a interrogação para o servidor, que faz a execução localmente e envia os resultados para o cliente. Quando existem múltiplos servidores, é necessária uma camada intermédia para lidar com a processamento de interrogações que envolvem dados localizados em diferentes servidores.

Esta abordagem é vantajosa quando o servidor tem um desempenho elevado e os clientes são lentos. Por outro lado, como ponto negativo, a sobrecarga do servidor está directamente relacionada com o número de clientes.

Na Figura 3.3, é ilustrado o processo para o caso de um servidor. Esta opção é usada em muitos SGBD relacionais (p.e. Oracle, DB2, SQL Server).

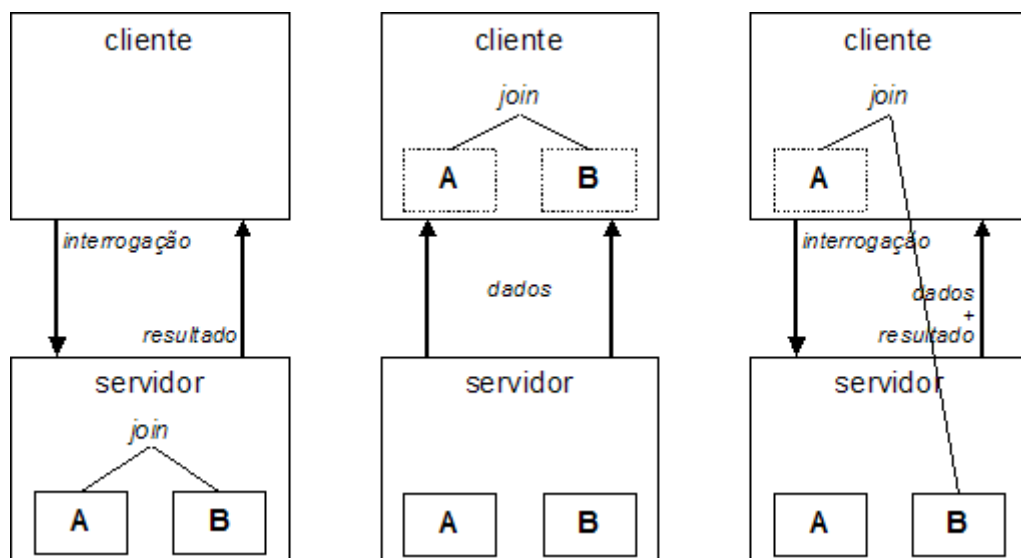


Figura 3.3: Colocação dos Dados: Envio da Interrogação, Envio dos Dados e Solução Híbrida

Colocação dos Dados: Envio da Interrogação, Envio dos Dados e Solução Híbrida
(imagem adaptada de [61])

3.5.2 Mover os Dados

Com o envio dos dados, todas as operações são executadas no cliente onde a interrogação teve origem. Funciona de forma oposta à abordagem anterior. O cliente mantém cópias dos dados existentes no servidor e utiliza estas cópias para processamento das interrogações. Neste caso, é necessário definir uma política de actualização das cópias que existem nos clientes.

Uma vantagem desta solução é o facto do processamento estar distribuído pelos clientes, permitindo um crescimento mais equilibrado. Do ponto de vista negativo, esta abordagem pode originar um volume de tráfego sobre a rede muito elevado.

No exemplo apresentado na Figura 3.3, cópias das tabelas A e B são mantidas no cliente e a interrogação (*join*) é executada sobre esses dados. Esta alternativa é usada em vários SGBD orientados a objectos (p.e. ObjectStore).

3.5.3 Solução Híbrida

Nenhuma das alternativas anteriores é a melhor para todos os casos. As vantagens de ambas podem ser combinadas numa solução híbrida [44]. Neste caso, a execução das interrogações pode ser feita no cliente ou no servidor e é possível a existência de *caches* nos clientes.

Esta abordagem é ilustrada na Figura 3.3. A interrogação inicial é dividida em duas parciais, uma é executada sobre a cópia local da tabela A e a outra é enviada para o servidor. O servidor executa a interrogação parcial e envia os resultados para o cliente, que reúne os dois conjuntos para produzir o resultado final.

3.6 Consistência dos Dados

Em sistemas distribuídos, a replicação de dados permite melhorar o tempo de resposta e a disponibilidade do sistema [16]. Por exemplo, numa situação em que os custos de comunicação sejam elevados, a replicação dos dados junto do cliente poderá ser uma opção vantajosa. Desta forma, para cada objecto *lógico*, podem existir várias representações *físicas* (Figura 3.4).

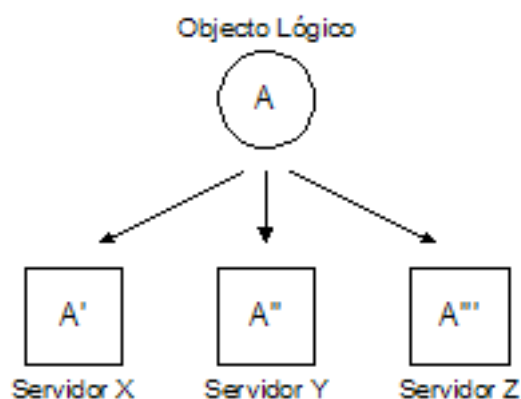


Figura 3.4: Representação Lógica e Representação Física

A existência de múltiplas representações físicas do mesmo objecto lógico introduz o problema da consistência dos dados. Como manter as cópias físicas iguais? Como processar as alterações feitas ao objecto lógico? É necessário um compromisso entre a consistência da informação e a performance e disponibilidade do sistema. Nas subsecções seguintes, apresentam-se as principais decisões associadas a uma arquitectura deste tipo.

Uma classificação exhaustiva de várias alternativas pode ser encontrada em Franklin [43]. Uma revisão interessante das opções nos principais produtos comerciais pode ser encontrada em Stacey [94].

3.6.1 Protocolos

Um protocolo especifica o processo segundo o qual modificações a uma cópia física são reproduzidas, de forma a restabelecer a consistência dos dados em todas as cópias físicas. Os protocolos respondem à questão: “Onde submeter as actualizações aos dados?”. Existem duas abordagens principais: *actualização no primário* e *actualização global* (Figura 3.5) [48].

Actualização no Primário

Neste caso, por vezes designado de *replicação passiva* [16], para cada objecto, é definida uma *cópia física primária*. As restantes cópias físicas são designadas por *cópias secundárias* ou *réplicas*. Este protocolo especifica que as operações de leitura podem ser feitas sobre qualquer uma das cópias, enquanto que as operações de escrita apenas podem ser realizadas sobre a cópia primária.

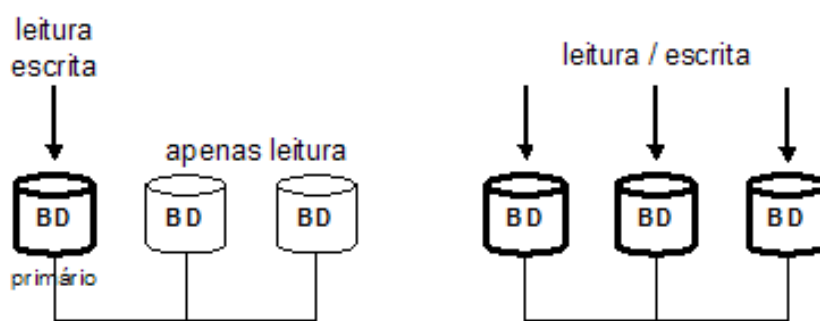


Figura 3.5: Protocolos de Replicação de Dados

As actualizações feitas sobre a cópia primária são depois propagadas para as réplicas (ver Subsecção 3.6.2).

A principal vantagem desta abordagem é a simplicidade de implementação. Os conflitos são detectados e resolvidos no primário. Do ponto de vista negativo, existe uma grande dependência do sistema primário, resultando numa menor tolerância a falhas.

Actualização Global

Nesta abordagem, também designada de *replicação activa*, as operações de leitura e escrita dos dados podem ser feitas sobre qualquer uma das cópias físicas. Assim, a manutenção da consistência é muito mais complexa porque, ao contrário do cenário anterior, não existe uma cópia primária. Todas as cópias físicas funcionam como primárias. A integridade dos dados não é garantida, uma vez que podem coexistir várias cópias físicas com um historial de transacções completamente independente.

A consistência é mantida com base em protocolos do tipo *replicação activa, cópias disponíveis* (ROWA) ou *baseados em quóruns* [97].

Este protocolo tem como vantagens um melhor balanceamento da carga e uma maior tolerância a falhas. Por outro lado, tem uma maior complexidade de implementação. A manutenção da consistência entre as diversas cópias exige protocolos sofisticados.

3.6.2 Propagação

Para a propagação da informação entre as cópias físicas, existem duas estratégias possíveis: *actualização síncrona* (*eager*) ou *actualização assíncrona* (*lazy*) (Figura 3.6) [48].

Síncrona

No caso da propagação síncrona, a alteração dos dados é feita em simultâneo nas diversas cópias físicas. As diversas actualizações são realizadas no contexto da mesma transacção, utilizando protocolos do tipo *two-phase commit*.

A propagação síncrona exige uma disponibilidade e resposta total por parte dos sistemas envolvidos e é, por isso, de muito difícil implementação em sistemas reais [76].

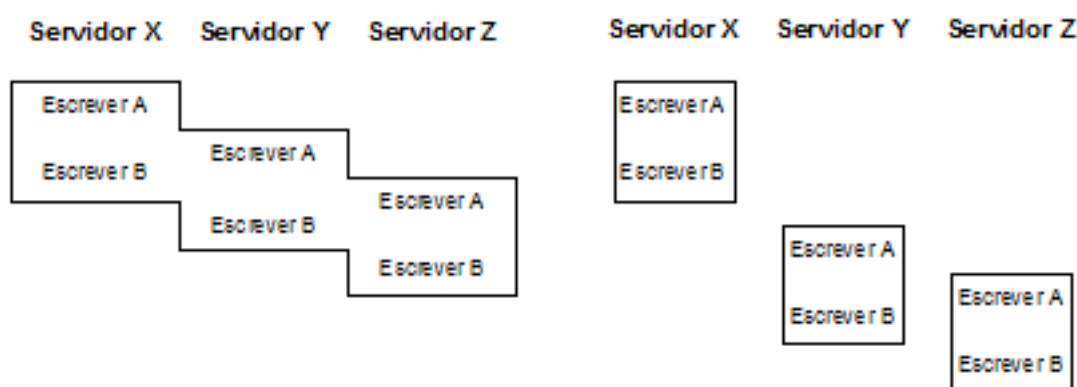


Figura 3.6: Propagação Síncrona e Propagação Assíncrona
Propagação Síncrona e Propagação Assíncrona (adaptado de [48])

As situações de bloqueio (*deadlocks*) e transacções abortadas são frequentes [48]. Como aspecto positivo, esta abordagem tem a vantagem de oferecer um elevado nível de consistência dos dados.

Assíncrona

No caso da propagação assíncrona, a transacção pode terminar após a actualização de uma das cópias. As outras réplicas são actualizadas de forma assíncrona e em transacções individuais.

Esta abordagem é mais tolerante a falhas e adequada a sistemas que se pretendem pouco ligados/dependentes (*loosely coupled*). Tem uma melhor resposta porque se evitam as esperas associadas a protocolos com múltiplas fases de finalização (*commit*). Do ponto de vista negativo, a *frescura* da informação é menor, pois está dependente de uma segunda fase de actualização.

3.6.3 Iniciativa

Uma terceira dimensão, associada à manutenção da consistência de informação distribuída, relaciona-se com a iniciativa da operação de actualização. As actualizações podem ser enviadas (*push*) ou pedidas (*pull*). Também existem soluções híbridas, que procuram activar dinamicamente uma das alternativas em função de critérios pré-estabelecidos (como o *rácio leitura-escrita*) [28].

Envio

Num cenário de envio, também designado por *baseado no servidor*, as actualizações são propagadas para as restantes réplicas sem serem pedidas. O servidor com a cópia mais recente mantém informação sobre as outras réplicas e, quando existe uma actualização, envia uma notificação. Esta notificação pode ser imediata ou não e de dois tipos: *invalidação* ou *actualização*.

No caso da **actualização**, é enviada a informação nova; enquanto que, no caso da **invalidação**, é enviada apenas uma indicação de que a informação foi alterada. Depois de receber uma invalidação, as réplicas pedem a informação nova. O envio de actualizações pode ser comparado com a propagação síncrona e o envio de notificações com a propagação assíncrona [72].

Numa arquitectura de envio dos dados, há um uso eficiente dos recursos. As interações ocorrem apenas quando há alterações. Outro aspecto positivo é o facto da consistência da informação ser elevada. As outras réplicas podem ser actualizadas assim que a primeira actualização acontece. É a melhor opção quando o *rácio leitura-escrita* é alto.

É, no entanto, uma solução mais complexa. Cada servidor tem de manter informação sobre o estado dos dados nas diversas réplicas e iniciar todas as actualizações. Há uma maior dependência entre os participantes e, do ponto de vista tecnológico, são necessárias interfaces mais sofisticadas. Com o aumento do número de réplicas, existe um aumento da carga sobre o servidor que efectua a actualização.

Pedido

Alternativamente, nesta abordagem também designada por *baseada no cliente*, as actualizações são pedidas pelas réplicas. A actualização acontece no momento do pedido. Uma estratégia comum é, por exemplo, o envio de pedidos com base numa periodicidade pré-estabelecida (um intervalo de *pooling*).

Neste caso, e ao contrário da situação anterior, a *informação de estado* que é necessário manter é significativamente menor. As réplicas apenas necessitam de saber qual o servidor que devem interrogar para obter a cópia mais recente. É uma abordagem mais eficiente quando o *rácio leitura-escrita* é baixo. Como as actualizações são frequentes, o número de pedidos de informação que não foi alterada é menor.

No geral, a arquitectura do sistema é mais simples. Existe um maior desacoplamento entre os participantes e uma melhor distribuição dos custos de processamento; o que resulta num maior potencial de crescimento.

Por outro lado, em muitas situações, há um desperdício significativo de recursos computacionais e de rede de comunicação. Em cenários onde os dados têm poucas actualizações, são vulgares os pedidos de itens que não sofreram qualquer alteração. Também como aspecto negativo, o nível de consistência da informação é menor. Varia em função da frequência de actualização pré-estabelecida (um valor difícil de calcular).

3.7 Entrega dos Dados

Um outra dimensão que permite classificar e distinguir sistemas distribuídos, está relacionada com o que se designa por *Esquemas de Entrega dos Dados*. Entrega dos dados pode ser definido como “o processo de entrega de informação de um conjunto de fontes de informação (servidores) para um conjunto de consumidores de informação (clientes)” [65].

É possível distinguir três características que podem ser usadas para a comparação entre mecanismos de entrega dos dados: protocolo, frequência e modo [42].

3.7.1 Protocolos

- **Pedido da Informação (*Pull*):** Neste caso, a transferência da informação é iniciada pelo cliente. Quando o pedido é recebido pelo servidor, a informação é reunida e é enviada apenas para o cliente que fez o pedido.
- **Envio da Informação (*Push*):** Neste protocolo, a interacção é iniciada pelo servidor. O servidor mantém informação sobre o conjunto de clientes e, para cada novo item de informação, faz o envio para todos ou um subconjunto dos clientes.

3.7.2 Frequência

- **Aperiódico:** Neste esquema, a interacção acontece em função de eventos que ocorrem. No cenário de pedido da informação (*pull*), a interacção pode ser despoletada por uma acção do utilizador. No cenário de envio da informação (*push*), uma actualização aos dados pode iniciar a transferência. Um exemplo desta situação é a consulta de informação dos mercados de valores feita por um utilizador.
- **Periódico:** Neste caso, a entrega dos dados ocorre segundo uma calendarização pré-definida. Este intervalo de tempo pode ser fixo ou incluir um factor aleatório. É o que se verifica no envio de notícias em intervalos regulares.

3.7.3 Modos

- **Envio Único (*Unicast*):** Neste caso, os dados são enviados uma única vez do servidor para o cliente.
- **Envios Múltiplos (*1-to-N*):** Neste caso, a mesma fonte pode enviar dados para múltiplos clientes (N). Existem dois subtipos de envios múltiplos: *multicast* e *broadcast*.

3.7.4 Classificação dos Esquemas

Na Figura 3.7, é apresentada uma classificação com base nas três características enumeradas antes. Nas folhas do diagrama estão identificados vários esquemas de entrega de dados.

3.8 Revisão das Tecnologias

Do ponto de vista tecnológico, as primeiras abordagens ao problema da interoperabilidade entre sistemas informáticos têm origem na década de 70; as primeiras especificações EDI surgiram na indústria dos transportes dos EUA. Nesta secção, faz-se uma breve apresentação de algumas das principais tecnologias; uma lista exhaustiva de soluções para a comunicação distribuída entre sistemas de informação empresariais pode ser encontrada em [67].

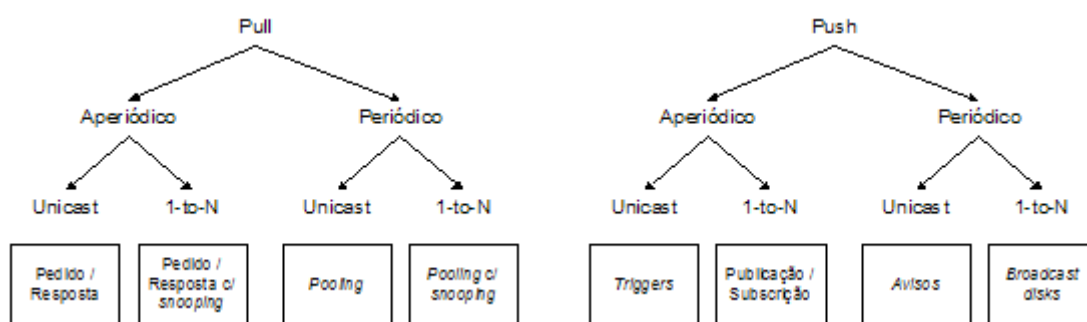


Figura 3.7: Esquemas de Entrega dos Dados
Esquemas de Entrega dos Dados (Adaptado de [42])

3.8.1 EDI

Electronic Data Interchange (EDI) é uma designação genérica, usada para referir os protocolos que permitem levar a cabo transacções electrónicas de documentos empresariais, entre parceiros. Na base do seu funcionamento, está o envio de mensagens fortemente estruturadas segundo normas (p.e.: ANSI X21 e UN/EDIFACT, ver Secção 2.4.2) e que podem ser processadas sem intervenção humana [67]. Ao nível da comunicação, são utilizadas redes de valor acrescentado (VAN) para a troca de mensagens entre os parceiros. Estas redes fornecem serviços como: autenticação, segurança, fiabilidade e integridade. Ver Figura 3.8 (adaptada de [67]).

Para além da sintaxe, as especificações EDI definem a semântica das mensagens trocadas. Refira-se, a título de exemplo, que o formato HL7 [54] foi criado por um conjunto de instituições ligadas à indústria da saúde, com o objectivo de estabelecer os procedimentos e fluxos para este domínio.

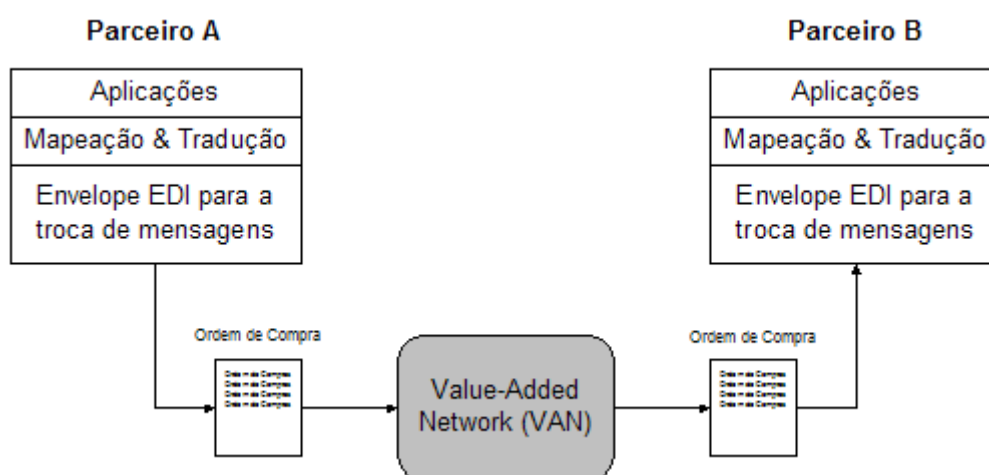


Figura 3.8: Arquitectura EDI

Os protocolos EDI estão bem estabelecidos na indústria na medida em que a infraes-

estrutura existe, as organizações de normas publicam e actualizam as especificações e, uma vez implementados, permitem aos parceiros participar na troca de mensagens [9]. No entanto, os custos associados à implementação destas soluções são elevados, em particular devido aos custos de subscrição das VAN e de instalação dos sistemas de integração [77]. As pequenas empresas ficam *de facto* excluídas destas parcerias. Nos EUA, 90% das empresas da Fortune 500 usam EDI, enquanto que das restantes 10 milhões apenas 6% o usam [67].

No sentido de reduzir o custo de uso associado às VAN, surgiram iniciativas que procuram implementar o EDI sobre a Internet, como o EDIINT e o OBI. Em [67] pode ser encontrada uma avaliação destas tecnologias.

3.8.2 Componentes

As evoluções tecnológicas deram origem a novas abordagens ao problema da interoperabilidade. Estas iniciativas podem ser agrupadas numa categoria mais abrangente designada por arquitecturas baseadas em componentes [67]. Componentes são módulos que podem ser associados a aplicações existentes ou desenvolvidos de raiz e permitem encapsular e disponibilizar funcionalidades. Podem ser implementados usando qualquer linguagem de programação e funcionam sobre infraestruturas (*middlewares*) que coordenam a comunicação entre os componentes.

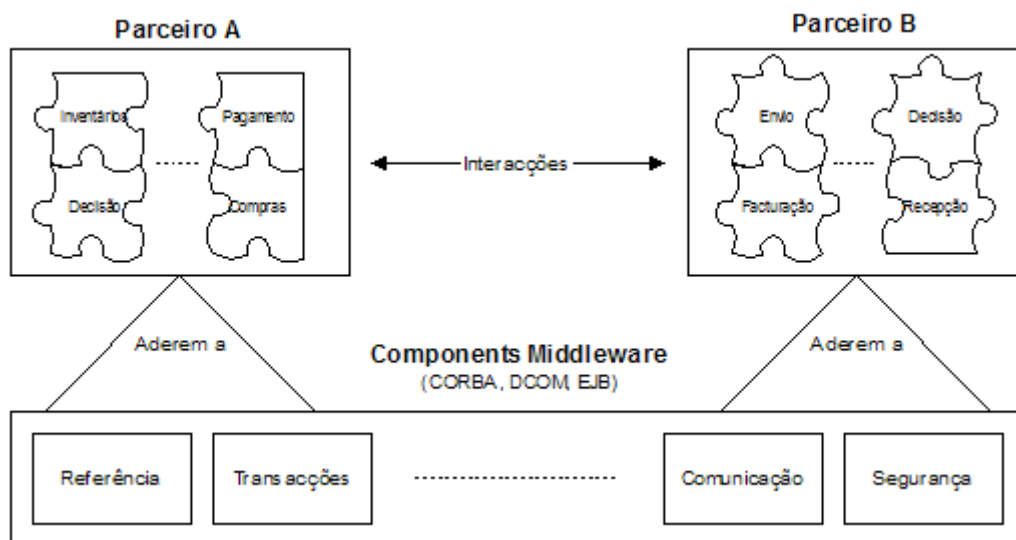


Figura 3.9: Arquitectura baseada em Componentes

A Figura 3.9 (adaptada de [67]) ilustra a estrutura geral de uma arquitectura baseada em componentes. Os módulos implementados pelos parceiros comunicam, usando a infraestrutura disponibilizada pela tecnologia. Essa infraestrutura oferece serviços como comunicação, mapeamento, segurança, transacções, etc.

As 3 mais importantes arquitecturas baseadas em componentes são [67]:

- Common Object Request Broker Architecture (CORBA)

CORBA é uma norma publicada pelo Object Management Group (OMG), um consórcio internacional da indústria [75]. O núcleo da infraestrutura são os ORB (*Object Request Broker*), que funcionam como nós intermediários e permitem a comunicação entre os diversos componentes. A complexidade técnica destas arquitecturas é elevada.

- **Distributed Component Object Model (DCOM)**
Semelhante ao CORBA, DCOM é a tecnologia da Microsoft para a comunicação distribuída baseada em componentes. Ao contrário do CORBA, que não está limitado ao nível da plataforma, o DCOM foi desenvolvido para funcionar sobre tecnologia Microsoft Windows (existem algumas implementações para outras plataformas).
- **Enterprise Java Beans (EJB)**
Na mesma linha das anteriores, as EJB são a aposta da Sun para a comunicação distribuída entre componentes. Ao nível da comunicação, esta arquitectura usa Java RMI. Apenas suporta Java como linguagem de desenvolvimento.

Uma descrição detalhada destas 3 tecnologias pode ser encontrada em [67].

Uma característica comum a estas tecnologias é o facto de se destinarem a ambientes fortemente acoplados e a relações longas. Enquanto que o EDI “ataca” o problema ao nível empresarial, com uma arquitectura baseada na troca de documentos, estas tecnologias funcionam ao nível das aplicações, resultando numa arquitectura mais “apertada”.

São soluções com muitas limitações em ambientes heterogéneos, dinâmicos e fortemente desacoplados [104], como o que se verifica no contexto dos SIU.

3.8.3 Serviços Web

Com o desenvolvimento da Internet e das tecnologias paralelas, surgiram novas abordagens ao problema da interoperabilidade entre sistemas de informação. A ubiquidade da “rede das redes”, “indiferente” a fronteiras empresariais ou tecnológicas, surge como uma alternativa para a camada de comunicação. Por outro lado, a evolução tecnológica e a adopção generalizada da *eXtensible Markup Language* (XML), como linguagem de estruturação de documentos, veio permitir o desenvolvimento de novas arquitecturas, actualmente designadas de Serviços Web (SW) (Figura 3.10).

Haas [52] define Serviço Web como “um sistema informático identificado por um URI, cujas interfaces públicas e ligações são definidas e descritas usando XML. Estas definições podem ser descobertas por outros sistemas informáticos. Estes sistemas podem depois interagir segundo as regras definidas, usando mensagens baseadas em XML, transportadas por protocolos Internet”.

Uma característica importante dos Serviços Web é o facto da comunicação se basear, tal como no EDI, na troca de documentos, resultando numa arquitectura muito desacoplada.

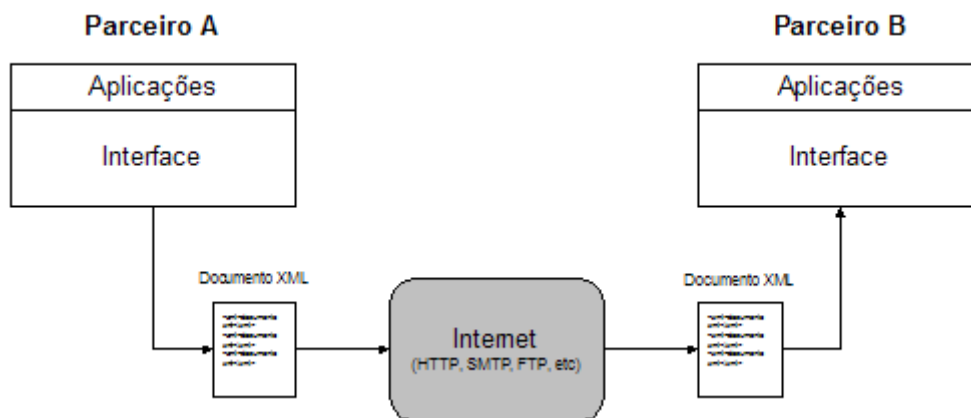


Figura 3.10: Arquitectura baseada em Serviços Web

A ideia de XML sobre a Internet é simples e poderosa, mas para uma real adopção no mundo empresarial existem questões que é necessário ultrapassar. Ao nível da comunicação, emergem questões como a segurança, a fiabilidade, a privacidade e a qualidade de serviço. Por outro lado, a linguagem XML permite apenas definir a sintaxe de um documento; ficam em aberto questões como a semântica e a especificação das sequências de mensagens numa comunicação empresarial.

Assim, com o intuito de resolver estes problemas, surgiram várias iniciativas que, através da publicação de recomendações e normas, procuram ultrapassar estas limitações. No entanto, nos últimos anos, tem-se assistido a uma multiplicação exagerada de normas e especificações que, frequentemente, se sobrepõem, donde resulta que a adopção destas tecnologias tem seguido um ritmo mais lento do que o esperado. A propósito, Bussler afirma que “a designação ‘norma’ já não corresponde a ‘definição global e única’” [9].

Apesar de tudo, os Serviços Web e as tecnologias associadas têm tido um forte apoio por parte das principais empresas da indústria dos computadores e de organizações independentes como o W3C, que tem contribuído activamente para o desenvolvimento de especificações nesta área.

Capítulo 4

Definição do Problema

Neste capítulo, é apresentado o problema concreto de interoperabilidade entre sistemas de informação universitários tendo como caso de estudo o cenário da Universidade do Porto.

Para a caracterização do problema foi feita uma estruturação do domínio em dois eixos. Segundo um eixo vertical, é abordada a ligação entre o nível faculdade e o nível universidade, tendo como principal objectivo o estudo dos processos de agregação, composição e disseminação de informação. Segundo o eixo horizontal analisa-se a partilha de informação entre faculdades, em particular informação sobre os alunos.

No final, são seleccionados e apresentados quatro cenários de aplicação, utilizados como casos de estudo. Para cada um dos casos, é feita uma breve contextualização, são apresentadas algumas situações reais de aplicação e são definidos os requisitos.

4.1 Apresentação do Contexto

A Universidade do Porto (UP) é uma universidade estatal, fundada a 22 de Março de 1911. É composta por 17 unidades orgânicas, 14 equiparadas a faculdades e 3 não equiparadas, todas com autonomia pedagógica, administrativa e financeira, reunindo, no seu conjunto, mais de 25 mil alunos [29].

Cada unidade gere de forma independente grande parte da informação de que dispõe. A gestão dos alunos, em particular, é feita localmente por cada uma das instituições.

A Reitoria da Universidade disponibiliza uma aplicação informática para a gestão de alunos (GAUP), que é usada pela maioria das instituições. O GAUP tem uma arquitectura distribuída. Cada instituição possui uma cópia da aplicação que se integra com uma base de dados local. Os dados existentes em cada instituição são, mensalmente ou sempre que necessário, copiados de forma integral para a reitoria, que actualiza uma base de dados central. Nos casos em que as instituições usam sistemas próprios, esta integração é efectuada manualmente e caso a caso.

Por outro lado, a gestão dos recursos humanos de toda a Universidade é feita de forma centralizada na Reitoria, usando uma aplicação desenvolvida internamente (GRHUP). As instituições não têm acesso directo aos dados.

Tendo presente os objectivos desta dissertação, procurámos identificar questões relacionadas com a interoperabilidade entre os diversos sistemas de informação no contexto universitário. Com base nos vários processos identificados e na natureza das organizações, definiram-se dois eixos de interoperabilidade: horizontal e vertical.

4.2 Eixos de Interoperabilidade

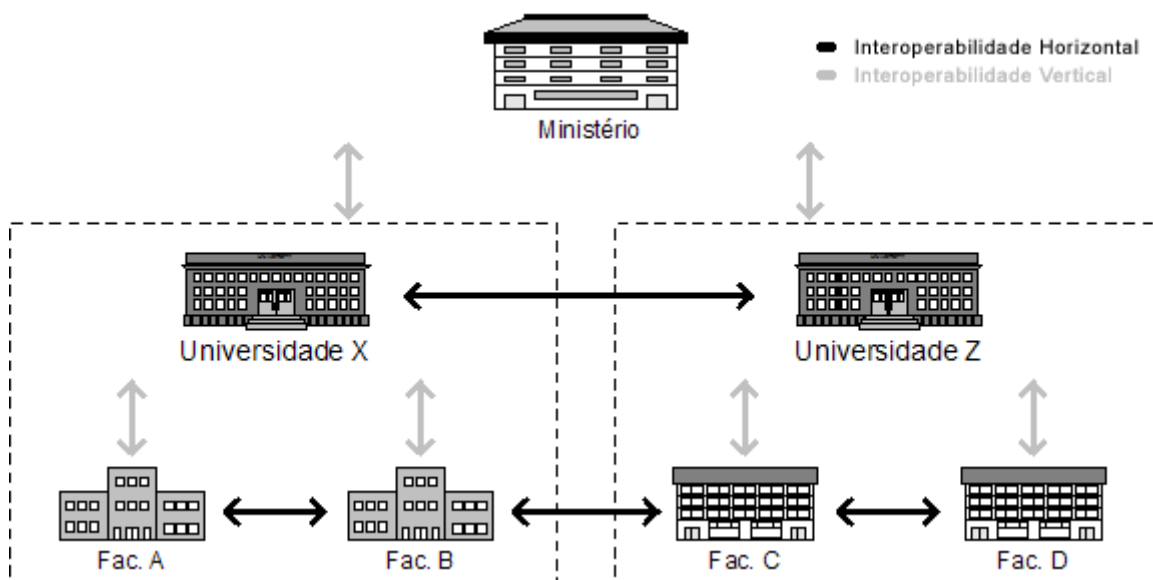


Figura 4.1: Interoperabilidade Vertical e Horizontal

4.2.1 Interoperabilidade Horizontal

A designação “interoperabilidade horizontal” identifica os fluxos entre as instituições académicas do mesmo nível administrativo. Referimos, a título de exemplo, a troca de informação relativa a um curso entre duas faculdades (ver Figura 4.1).

É possível enumerar um conjunto de cenários, que exemplificam a importância da interoperabilidade a este nível entre SIU:

- A cooperação entre instituições é cada vez mais frequente, nomeadamente na implementação de cursos ou projectos multi-disciplinares. Neste contexto, a **partilha de dados** de uma forma automática constitui um factor importante. Um exemplo concreto é o dos cursos partilhados por várias instituições, em que a informação sobre os conteúdos das disciplinas deve existir nos diversos SI.
- A possibilidade de mobilidade das pessoas dentro do espaço académico europeu é um dos principais objectivos resultantes da declaração de Bolonha [36]. A integração de **alunos do programa ERASMUS** é um exemplo da necessidade de troca de informação entre instituições heterogéneas.
- No contexto dos **núcleos de investigação e desenvolvimento**, a cooperação entre diferentes instituições é vulgar e, muitas vezes, necessária. Nestes ambientes, a comunicação e a partilha de documentos entre as partes envolvidas não pode constituir um obstáculo. Aliás, foi esta a motivação para o desenvolvimento da World Wide Web, um projecto que envolvia a partilha de informação entre investigadores de diferentes laboratórios do CERN [4].

4.2.2 Interoperabilidade Vertical

Interoperabilidade vertical corresponde aos fluxos existentes entre as instituições académicas em níveis administrativos diferentes. Por exemplo as trocas de informação entre uma faculdade e os órgãos centrais da universidade, ou entre um ministério e a reitoria de uma universidade. (Figura 4.1).

No meio académico, os fluxos segundo o eixo vertical são mais frequentes do que os existentes entre instituições do mesmo nível. Este padrão resulta da natureza hierárquica, mas descentralizada, das estruturas académicas. O poder está disperso por unidades independentes, que recorrem a órgãos centrais com funções administrativas e coordenadoras (ver Capítulo 2).

No caso concreto dos sistemas de informação a este nível, podemos distinguir um conjunto de cenários em que a interoperabilidade é importante:

- A **concentração de informação** é uma necessidade frequente do ponto de vista da universidade. A existência de sistemas heterogéneos dispersos pelas faculdades torna a tarefa complexa. É, por isso, evidente a importância da existência de mecanismos que permitam consultar de forma transversal todos os sistemas das diversas faculdades e que se revelam indispensáveis para responder a perguntas como “Quais os endereços de correio electrónico de todos os alunos da universidade?”, ou simplesmente para obter a lista de “todos os docentes da universidade”.

- As tarefas de **gestão administrativa de recursos humanos ou físicos** são muitas vezes centralizadas: a gestão do pessoal docente e não docente, assim como a de certos equipamentos é vulgarmente efectuada ao nível da universidade. No entanto, a informação sobre estes recursos deve existir de forma partilhada em ambos os sistemas de informação, da faculdade e da universidade.
- A concorrência crescente, os cortes orçamentais e a uniformização do espaço académico europeu constituem um incentivo para a aposta na qualidade. Esta aposta passa frequentemente pela implementação do **cálculo de métricas associadas aos cursos**. Nesse sentido, o acesso a este tipo de informação, dispersa pelas diversas unidades da universidade, é crucial.
- A **agregação de informação** a nível nacional constitui outro exemplo de interoperabilidade vertical. Em Portugal, o Observatório da Ciência e do Ensino Superior [25] tem um projecto que visa a constituição de uma base de dados nacional com informação sobre as actividades académicas desenvolvidas nas universidades portuguesas. A existência de políticas e mecanismos de interoperabilidade definidos entre os órgãos de gestão central e as universidades permitiria agilizar este processo.

4.3 Selecção de Casos de Estudo

Nesta secção, descrevemos quatro cenários de aplicação, utilizados como casos de estudo. Para o eixo horizontal, foi identificada a partilha do registo académico do aluno como aplicação real e exemplificativa das interacções existentes entre instituições ao mesmo nível (Subsecção 4.3.2).

No caso do eixo vertical, foram identificados três tipos de interacções entre instituições de níveis diferentes: agregação, concentração e difusão de informação.

A agregação de informação existe nos cenários em que informação normalmente dispersa é resumida num ponto central (p.e. num documento). Na Subsecção 4.3.3, e como exemplo de agregação para o eixo vertical, apresenta-se o caso da compilação de estatísticas relativas aos cursos e disciplinas.

A concentração de informação é semelhante à agregação, mas neste caso pretende-se replicar centralmente informação dispersa por várias instituições e não resumi-la. Neste caso, será necessário considerar a coordenação das várias entidades envolvidas. Na Subsecção 4.3.4 é apresentada a concentração de informação sobre os recursos humanos de toda a universidade como exemplo deste tipo de interacção.

A difusão de informação corresponde a interacções no sentido contrário, ou seja, da Universidade para as faculdades. Como caso de estudo refere-se a difusão de notícias, produzidas na Reitoria, por toda a comunidade universitária (Subsecção 4.3.5).

4.3.1 Requisitos Comuns

Nesta secção, apresentam-se os requisitos comuns a todos os casos de estudo seleccionados. Cada requisito tem uma referência única composta por um código e um número. Os códigos têm o seguinte significado: **RF** (Requisito Comum Funcional), **RNF** (Requisito Comum Não Funcional), **RHF** (Requisito Horizontal Funcional) e **RV1F**, **RV2F** e **RV3F**

(Requisitos Verticais Funcionais). Os requisitos verticais estão estruturados em função dos três casos de estudo.

Os requisitos particulares de cada caso de estudo são apresentados junto do caso de estudo. A ordem pela qual os requisitos são apresentados não reflecte nenhum grau de importância.

Requisito RF1 : A autenticidade dos documentos deve ser garantida

Descrição :

A especificação final deve incluir a descrição dos procedimentos a seguir, por cada uma das partes, de forma a garantir a autenticidade da informação trocada.

Requisito RF2 : A integridade da informação deve ser garantida

Descrição :

A especificação final deve incluir a descrição dos procedimentos a seguir, por cada uma das partes, de forma a garantir a integridade da informação trocada.

Requisito RF3 : Deve ser possível garantir a confidencialidade de toda ou parte da informação

Descrição :

A especificação final deve incluir a descrição dos procedimentos a seguir, por cada uma das partes, de forma a garantir a confidencialidade da informação trocada. Os intervenientes na comunicação devem poder decidir que parte da informação cifrar.

Requisito RF4 : Qualquer idioma pode ser usado na informação transmitida

Descrição :

A informação transmitida deve poder ser representada em qualquer idioma.

Justificação :

Reduzir os custos associados à adopção desta solução por parte de outras instituições, portuguesas ou de outros países.

Requisito RNF1 : Não usar tecnologias fechadas na implementação**Descrição :**

As tecnologias escolhidas para a implementação devem ser independentes de fabricantes, linguagens de programação ou equipamentos. As soluções escolhidas devem ser baseadas em normas acreditadas por organizações nacionais ou internacionais de normas, ou por consórcios industriais relevantes (como o W3C).

Justificação :

Eliminar dependências face a fabricantes, tecnologias ou equipamentos. Reduzir o risco associado à adopção das tecnologias.

Requisito RNF2 : Apresentar especificações formais para as soluções propostas**Descrição :**

Devem ser desenvolvidas especificações formais para a representação da informação e para os mecanismos de interacção propostos nos diversos casos de estudo.

Justificação :

A existência de uma especificação aprovada serve de “contrato” entre as instituições aderentes e pode ser usada para validar e certificar as diversas implementações.

Requisito RNF3 : Os termos de controlo usados na codificação e comunicação devem ser em Inglês**Descrição :**

Os termos de controlo, usados na codificação ou transporte da informação, devem ser representados usando o idioma Inglês.

Justificação :

O carácter internacional do projecto, em concreto a troca de registos académicos no âmbito de programas de mobilidade internacional, justificam esta opção.

Requisito RNF4 : Funcionar, em situações normais, com as condições de conectividade existentes**Descrição :**

A solução proposta deve fazer uso das condições de conectividade actualmente existentes entre as instituições. As tecnologias escolhidas devem ser, preferencialmente, independentes da camada de transporte.

Justificação :

Evitar soluções dependentes de redes ou protocolos privados, controlados por empresas individuais ou pequenos grupos. A ubiquidade da Internet e os baixos custos de instalação e utilização justificam a opção por esta via.

4.3.2 Partilha do Registo Académico do Aluno

Definição do Contexto

O Registo Académico do Aluno reúne informação sobre todas as actividades académicas de cada aluno, ao longo do seu percurso no ensino superior. Este registo acompanha-o sempre e é usado, nomeadamente, para enviar informação sobre a sua situação em caso de transferência entre instituições académicas. Geralmente, este registo não existe sob a forma de documento único; está disperso por vários documentos em diferentes serviços.

A existência de um documento electrónico, escrito de acordo com regras bem definidas, que inclua todos os dados considerados essenciais, permite normalizar e automatizar o processo de transferência de informação. A normalização acontece porque existe uma especificação para o registo académico do aluno que funciona como “gramática” e permite o mesmo entendimento por parte de parceiros diferentes. A automatização é possível porque, devido à existência deste documento “bem definido”, os dados podem ser processados sem intervenção humana. A automatização pressupõe uma normalização prévia.

A informação sobre o aluno inclui:

1. **Informação Demográfica:** dados pessoais e identificadores.
2. **Informação sobre Saúde:** estado das vacinas e exames médicos.
3. **Informação sobre a Situação Militar:** situação militar.
4. **Informação Académica:**
 - (a) **Ingresso:** informação sobre as provas realizadas no âmbito do acesso ao ensino superior.
 - (b) **Historial Académico:** as instituições frequentadas, os graus atingidos, as disciplinas e os resultados.
 - (c) **Prémios Académicos:** informação sobre prémios académicos atribuídos.

5. **Actividades Extra-curriculares:** informação sobre actividades extra-curriculares (projectos, associações, cursos).
6. **Outras Informações:** informações extra, relevantes para o registo académico.

Cenários de Aplicação

Troca de registos entre faculdades De forma análoga à que se verifica com o registo académico do aluno em papel, as escolas podem solicitar o envio de registos de alunos em formato electrónico. Este pedido pode efectuar-se durante o processo de transferência do aluno e o envio do documento electrónico pode ser feito em suporte físico (disco) ou por transferência electrónica (Internet). O registo académico do aluno pode ser gerado de forma automática por um sistema de informação ou produzido manualmente, de acordo com a especificação.

Troca de registos no contexto de programas de mobilidade internacional Durante um processo de intercâmbio enquadrado nos programas de mobilidade (p.e. ERASMUS), existe necessidade de trocar, entre as faculdades envolvidas, informações sobre os alunos. Esta informação é um subconjunto daquela que é necessária para o registo académico completo do aluno.

Considerações

Nesta subsecção, apresenta-se um conjunto de questões associadas à proposta de um formato normalizado para a transferência de informação sobre os alunos. Os processos de decisão relativos a cada um dos pontos em análise serão descritos aquando da implementação da solução.

Identificação do Aluno Os sistemas de informação necessitam de identificadores únicos para representar pessoas ou documentos. No caso do registo académico do aluno, é necessário definir uma forma de identificar univocamente cada estudante.

Estes identificadores devem funcionar num contexto, preferencialmente, universal. A troca de registos sobre alunos pode existir entre instituições de um mesmo país ou de países diferentes. Colocam-se várias questões (embora algumas delas ultrapassem o âmbito desta dissertação):

- Quem deve ser responsável pela definição e atribuição dos identificadores?
- O identificador deve ser privado ou pode ser disponibilizado publicamente?
- Qual o universo a que se destina este identificador: que alunos, que instituições, que graus e que países?
- O identificador é permanente ou pode ser alterado?

Graus Académicos No contexto nacional, existe uniformidade na definição dos graus académicos, as instituições regem-se por regras definidas pelo governo central. No entanto, quando se toma em consideração o âmbito internacional, é necessário ter em atenção as diferenças entre os vários sistemas de educação, em particular as equivalências entre os graus académicos. A declaração de Bolonha [36], com data de conclusão dos trabalhos fixada para 2010, inclui entre os principais objectivos a uniformização do ensino superior na Europa.

Uma das medidas consideradas como mais importantes neste cenário é a adopção de um sistema de créditos comum - *Sistema Europeu de Transferência e Acumulação de Créditos* (ECTS¹) [37].

Planos de Estudos Cada disciplina frequentada está integrada num determinado plano de estudos. Os planos de estudo estão associados aos cursos e são alvo de revisões periódicas, que podem corresponder à inclusão de novas disciplinas ou à eliminação de outras. Assim, quando se comparam estudantes com o mesmo curso da mesma faculdade, é possível que o plano de estudos frequentado seja diferente.

No plano de estudos de cada curso, são definidos os conteúdos programáticos e os créditos de cada uma das disciplinas que o integram. Quando se faz referência ao registo académico de aluno, é imprescindível considerar a informação sobre o plano de estudos em que se enquadram as disciplinas frequentadas. Também neste contexto se colocam algumas questões:

- O plano de estudos deve ser anexado integralmente ao registo académico do aluno ou apenas referenciado?
- Quem deve ser responsável pela disponibilização de informação sobre todos os planos de estudo de um curso?

Classificações No registo académico do aluno é incluída informação sobre a classificação obtida em cada disciplina. Esta informação é apresentada segundo uma escala, que varia de país para país (em Portugal de 0 a 20). Sem mais informações, é impossível comparar classificações absolutas provenientes de diferentes contextos (faculdades, países). Acresce ainda que, apenas com esta informação, é difícil a atribuição de equivalências entre cursos ou instituições.

Assim, para além da classificação absoluta, o registo académico do aluno deve integrar informação que permita enquadrar esta classificação num contexto mais abrangente. O sistema de créditos ECTS propõe a construção de percentis para ultrapassar estes problemas.

Equivalências O registo académico do aluno é vulgarmente utilizado para transferir informação entre instituições tendo em vista a possibilidade de atribuição de equivalências entre disciplinas. Nesta situação, há questões (algumas já mencionadas), a considerar:

- É necessário distinguir alunos com percursos diferentes mas com conclusão do curso na mesma instituição?

¹Do Inglês *European Credit Transfer System*.

- Quando são atribuídas equivalências para uma determinada disciplina, a classificação inicial é eliminada?
- Como gerir equivalências múltiplas, quando uma disciplina é equivalente a mais do que uma? E quando acontece o caso contrário?
- O que fazer com disciplinas que habitualmente não têm classificação quantitativa?

Internacionalização A troca de registos a nível internacional, em particular no âmbito dos programas de mobilidade, implica uma reflexão cuidada sobre outros problemas a ter em conta:

- Que idioma deve ser usado para a meta-informação e que mecanismos de transmissão?
- Deve haver distinção entre os registos usados no contexto nacional e os que são usados entre países? Deve ser usado o mesmo formato em ambas as circunstâncias ou os registos devem ser diferentes?
- Os registos enviados no âmbito do programa ERASMUS terão requisitos demasiado específicos, não contemplados nos registos usados internamente?
- A informação incluída nos registos nacionais, irrelevante num contexto internacional, deve ser removida dos registos transmitidos?

Segurança A segurança assume um papel importante num sistema de natureza distribuída. Neste caso, destacam-se, em particular, três variáveis: autenticidade, integridade e confidencialidade:

- Como garantir a veracidade da informação existente num Registo Académico do Aluno? Como distinguir os documentos genuínos dos fraudulentos?
- Como certificar que a informação consumida é igual à informação produzida? Isto é, como impedir a existência de erros na mensagem (acidentais ou intencionais)?
- O conteúdo do Registo Académico do Aluno deve ser confidencial? É necessário implementar mecanismos que permitam controlar o acesso à informação?
- Do ponto de vista da confidencialidade, o Registo Académico do Aluno deve ser tratado de forma homogénea? Há informação mais sensível do que outra?

Requisitos

Nesta secção, apresentam-se os requisitos para o cenário de interoperabilidade horizontal. Os requisitos estão agrupados em duas categorias: requisitos funcionais e requisitos não funcionais.

Requisito RHF1 : Identificar cada registo de forma universal**Descrição :**

Deve ser possível identificar de forma única cada registo académico. Esta identificação deve ser válida no contexto internacional.

Justificação :

O documento que representa o registo académico do aluno pode existir fora dos sistemas de informação (p.e.: suporte magnético), assim deve ser possível identificar de forma única a entidade responsável pela informação e o aluno a que se refere a informação. A possibilidade de existir troca de registos de aluno entre países, origina a necessidade de manter a identificação válida entre países.

Requisito RHF2 : Representar informação demográfica sobre o aluno**Descrição :**

O registo académico do aluno deve incluir:

- Nome completo;
- Número do Bilhete de Identidade;
- Local e data de emissão do Bilhete de Identidade;
- Local e data de nascimento;
- Nacionalidade;
- Género;
- Estado Civil;
- Filiação (*Incluir, para cada um dos pais, informação demográfica básica.*);
- Morada;
- Telefone;
- Telemóvel;
- Correio electrónico;
- Página Web.

Requisito RHF3 : Representar informação sobre a situação militar do aluno

Descrição :

O registo académico do aluno deve incluir informação sobre o estado da situação militar do aluno, quando aplicável.

Requisito RHF4 : Representar informação sobre o registo de saúde do aluno

Descrição :

O registo académico do aluno deve incluir informação sobre as vacinas administradas e respectiva validade, bem como os resultados dos exames médicos necessários à frequência do ensino superior.

Requisito RHF5 : Representar informação sobre as provas de ingresso efectuadas pelo aluno

Descrição :

O registo académico do aluno deve incluir informação sobre as provas de ingresso realizadas para acesso ao ensino superior.

Requisito RHF6 : Representar informação académica sobre o aluno

Descrição :

O registo académico do aluno deve incluir:

- Matrículas e inscrições efectuadas pelo aluno, com informação sobre o código interno, o ano, a instituição, o curso e o plano de estudos;
- Disciplinas concluídas com informação absoluta e também relativa sobre os resultados obtidos;
- Graus académicos obtidos.

Requisito RHF7 : Representar informação sobre prémios académicos obtidos pelo aluno

Descrição :

Sobre cada prémio académico incluir: a instituição responsável pela atribuição, o título, a data e o nível do prémio.

Requisito RHF8 : Representar informação sobre actividades extra-curriculares frequentadas pelo aluno

Descrição :

Sobre cada actividade extracurricular incluir: o título, uma descrição, a data de início e a data de fim.

Requisito RHF9 : Permitir representar outra informação, não especificada, sobre o aluno

Descrição :

O registo académico do aluno deve permitir incluir informação não especificada sobre o aluno (p.e. observações).

4.3.3 Agregação de Estatísticas sobre os Cursos

Definição do Contexto

O acesso a dados estatísticos relativos ao funcionamento das faculdades, por parte dos órgãos centrais da Universidade, é uma necessidade fundamental para a tomada de decisão. Neste caso de estudo, aborda-se a publicação de estatísticas pelas faculdades, relativas ao desempenho dos cursos, e a posterior agregação desta informação pela Reitoria da Universidade.

Cenários de Aplicação

Análise comparativa de desempenho A existência de informação relativa ao desempenho dos cursos e disciplinas num formato normalizado permite comparações objectivas. Estas comparações podem ser feitas entre cursos e disciplinas da mesma instituição ou de instituições diferentes, ao nível da Universidade.

Agregação de informação ao nível da Universidade Com as diversas faculdades a publicarem informação estatística sobre os cursos e disciplinas, é possível, ao nível da Universidade, agregar estes dados. Torna-se viável analisar de forma centralizada informação tradicionalmente dispersa. Por outro lado, com base nestes dados, é possível produzir **nova** informação. Por exemplo, com base nas estatísticas individuais dos cursos e disciplinas disponibilizadas pelas faculdades, é possível produzir na Reitoria, estatísticas de desempenho sobre grupos de faculdades.

Considerações

Granularidade dos Dados O nível de agregação dos dados pode ser mais ou menos detalhado. Por exemplo, é possível ir até ao detalhe do aluno (reunindo informação sobre a nota de cada aluno), ou, pelo contrário, armazenar apenas informação relativa à média dos finalistas do curso. É necessário apurar que tipo de estatísticas se pretende calcular

ao nível da Universidade e disponibilizar a informação necessária. Os relatórios DIMAS solicitados, pelo Ministério da Educação, às instituições de ensino superior no final de cada ano lectivo, são um bom ponto de partida.

Normalização dos Dados Um dos objectivos associado à compilação de estatísticas sobre os cursos é o cálculo de métricas tendo em vista a análise comparativa entre diferentes realidades. É, pois, importante ter em atenção a utilização de dados normalizados, passíveis de serem comparáveis. Depois de definir qual a informação a incluir, é necessário proceder à avaliação da necessidade de normalização para cada um destes valores.

Segurança Também neste caso é importante considerar a segurança dos dados segundo as 3 vertentes já enumeradas na Subsecção 4.3.2: autenticidade, integridade e confidencialidade.

Requisitos

Nesta secção apresentam-se os requisitos para o caso de agregação de informação, no contexto da interoperabilidade vertical.

Requisito RV1F1 : Identificar cada curso de forma universal

Descrição :

Deve ser possível identificar de forma única cada curso.

Requisito RV1F2 : Identificar o período temporal da informação representada

Descrição :

A informação incluída sobre o curso é relativa a uma edição em particular, que é necessário explicitar.

Requisito RV1F3 : Representar informação sobre o curso**Descrição :**

Sobre cada edição do curso, deve existir a seguinte informação:

- Código de identificação interno;
- Sigla;
- Nome;
- Número total de docentes por género;
- *Numerus Clausus*;
- Número de novos alunos;
- Média, mediana, desvio padrão, mínimo e máximo das classificações de candidatura dos novos alunos;
- Número de alunos que concluíram o curso;
- Média, mediana, desvio padrão, mínimo e máximo das classificações dos alunos que concluíram o curso.

4.3.4 Concentração de Informação sobre os Recursos Humanos**Definição do Contexto**

Apesar dos contratos de trabalho serem geridos ao nível da Reitoria, a restante informação relativa aos docentes e funcionários é reunida e mantida pelas próprias instituições. Assim, são normais os fluxos de informação no sentido da Reitoria para a concentração desta informação. Por exemplo, para a obtenção dos contactos de todos os docentes.

Neste caso de estudo, discute-se a construção de mecanismos de interoperabilidade, entre os sistemas de informação que permitam normalizar e automatizar estes processos de concentração de informação.

Cenários de Aplicação

Consulta de informação disponível nas faculdades A existência de mecanismos normalizados para acesso à informação dispersa pelas faculdades elimina a necessidade de tratar isoladamente a ligação com cada uma das faculdades. Na Reitoria da Universidade, torna-se possível consultar a informação disponibilizada por cada instituição, independentemente das infraestruturas e tecnologias usadas. A informação existente nas faculdades, sobre os docentes ou funcionários, pode assim ser uniformemente acessível do exterior, sujeita às restrições de acesso aplicáveis.

Publicação de informação ao nível da Universidade A concentração de informação actualizada na Reitoria da Universidade permite a construção de novos sistemas que façam uso dessa informação e a disponibilizem sob outras formas. Um exemplo possível é a apresentação desta informação no sítio Web da Universidade. Apesar da informação ser gerida ao nível das faculdades, é possível, desta forma, concentrá-la e apresentá-la a um nível central.

Considerações

Coordenação e Integração Neste cenário, o acesso aos dados é feito em tempo real. Ao contrário do que acontece nos outros casos, a informação está permanentemente disponível e é trocada de forma automática entre as instituições. Na concentração de informação, a coordenação assume um papel importante, nomeadamente nos aspectos seguintes:

- Como lidar com falhas ou erros de acesso aos dados de uma instituição? Os dados centralizados devem continuar disponíveis, mesmo desactualizados, ou deve o sistema ser interrompido?
- Até que nível os sistemas de informação devem ser capazes de descobrir e utilizar as interfaces sem intervenção humana?
- A informação sobre como aceder às interfaces das faculdades deve estar centralizada na Universidade? Quem é responsável pela sua manutenção?

Segurança Neste caso, a segurança também é importante, as 3 vertentes já enumeradas nos casos de estudo anteriores devem ser avaliadas (ver Subsecção 4.3.2).

Requisitos

Nesta secção, apresentam-se os requisitos para o caso da concentração de informação, no contexto da interoperabilidade vertical.

Requisito RV2F1 : Identificar cada funcionário de forma universal

Descrição :

Deve ser possível identificar de forma única cada funcionário.

Requisito RV2F2 : Representar informação sobre o funcionário

Descrição :

Sobre os funcionários, deve existir a seguinte informação:

- Sigla;
- Nome completo;
- Número do Bilhete de Identidade;
- Local e data de emissão do Bilhete de Identidade;
- Data e local de nascimento;
- Naturalidade;
- Género;
- Instituição a que pertence;
- Categoria;
- Estado;
- Contacto(s).

Requisito RV2F3 : Possibilitar o acesso à informação através de interfaces acessíveis por sistemas informáticos

Descrição :

A informação sobre cada funcionário deve estar disponível para consulta através de interfaces acessíveis por sistemas informáticos.

Justificação :

Permitir automatizar a concentração de informação, sem necessidade de intervenção humana.

4.3.5 Difusão de Notícias da Universidade

Definição do Contexto

Nos cenários anteriores referentes à agregação e concentração, os fluxos de informação são no sentido da Reitoria. A informação existe nas faculdades e é consumida pela Reitoria da Universidade.

No caso da difusão, os fluxos são no sentido contrário, a informação é produzida pela Reitoria e é destinada às faculdades, bem como à comunidade universitária em geral. Esta disseminação de informação, a partir da Reitoria da Universidade, permite, nomea-

damente, a distribuição de legislação pelas diversas faculdades. Neste caso, são estudados os mecanismos de distribuição de notícias produzidas na Reitoria por toda a Universidade.

Cenários de Aplicação

Disponibilização controlada de notícias A existência de um sistema para a disseminação de informação a partir da Reitoria para toda a Universidade, permite simplificar e agilizar o processo de distribuição das notícias produzidas centralmente. Esta distribuição pode ser feita de forma controlada, entre instituições seleccionadas; ou publicamente, disponibilizando e facilitando o acesso por parte de sistemas externos, sem qualquer vínculo à Universidade.

Considerações

Falhas Neste processo de disseminação e recepção de notícias, a intervenção humana será reduzida, sendo, portanto, imprescindível avaliar os procedimentos a seguir em eventuais falhas. Por exemplo, quais devem ser os procedimentos quando houver falhas na recepção das notícias? A Reitoria deverá garantir a entrega, mesmo que mais tarde?

Coordenação Como deve ser implementada a coordenação entre a Reitoria e as faculdades? Deve ser o nó central a enviar notificações sempre que existem notícias novas, ou devem ser os clientes a consultar periodicamente o sistema central à procura de actualizações? É importante avaliar as interrupções originadas por cada uma destas opções.

Filtragem Numa arquitectura baseada na difusão de informação, o conteúdo é produzido centralmente e os vários clientes “consomem” apenas parte da informação (a que for relevante para cada um). Assim, no caso das notícias difundidas pela Reitoria, pondera-se:

- A filtragem da informação deve ser feita pela Reitoria ou pelos clientes?
- Caso seja a Reitoria a gerir a filtragem da informação, como é que os clientes comunicam os seus “interesses”?
- A utilização de categorias definidas centralmente será uma solução? Neste caso, como serão mantidas estas categorias?

Segurança De forma análoga ao que foi feito nos casos anteriores, é necessário avaliar a segurança segundo os 3 eixos já enumerados (ver Subsecção 4.3.2).

Requisitos

Nesta secção, apresentam-se os requisitos definidos para o caso da difusão de informação, no contexto da interoperabilidade vertical.

Requisito RV3F1 : Permitir o acesso, por parte das faculdades, às notícias publicadas pela Reitoria

Descrição :

As faculdades devem poder consultar as notícias publicadas pela Reitoria.

Requisito RV3F2 : Representar informação sobre cada notícia

Descrição :

Relativamente a cada uma das notícias deve estar disponível a seguinte informação:

- Data de publicação e expiração;
- Título;
- Corpo;
- Nome do autor/emissor;
- Ligação directa para uma página Web correspondente a esta notícia.

Capítulo 5

Arquitectura

Neste capítulo, é definida a arquitectura do sistema. Na primeira secção (5.1), é apresentada a arquitectura para o cenário de interoperabilidade horizontal. No caso da interoperabilidade vertical (Secção 5.2), são descritas as opções tomadas para cada um dos cenários definidos: agregação, concentração e difusão de informação.

5.1 Interoperabilidade Horizontal

Introdução

No caso da interoperabilidade horizontal, existe uma grande diversidade de cenários possíveis; desde simples arquitecturas de troca de informação pontual entre instituições, até complexos sistemas de partilha de informação em tempo real entre faculdades de vários países. São vários os parâmetros que podem influenciar as opções ao nível da arquitectura do sistema, em particular dos mecanismos de partilha.

Assim, não se incluem, aqui, decisões relativamente à interoperabilidade horizontal de um ponto de vista genérico. Uma análise geral sobre as opções tecnológicas em ambientes distribuídos pode ser encontrada no Capítulo 3.

Aqui, é analisada a situação concreta de **partilha de informação sobre alunos, entre faculdades**. Pretende-se, neste caso, um sistema que permita disponibilizar de forma integrada e, em vários pontos, informação sobre um conjunto de alunos. Procura-se *concentrar em vários pontos* esse conjunto de dados.

Neste caso, existe um servidor - a instituição que detém informação sobre os alunos e vários clientes - as instituições que pretendem partilhar esta informação. Em relação à informação, há um mapeamento de *um-para-muitos*. A informação sobre um estudante, existente numa instituição, é partilhada em vários locais. No entanto, este padrão não corresponde a uma situação de difusão de informação, já que a informação é permanente e os clientes podem actualizá-la.

Colocação dos Dados

Relativamente à estratégia de colocação dos dados e neste caso, há uma correspondência directa entre os dados nos servidores e os dados no cliente. Por outro lado, sobre os mesmos dados podem ser realizadas várias operações de consulta. Assim, opta-se por *manter cópias da informação nas faculdades*, permitindo a existência de várias representações físicas de cada registo de aluno.

Consistência dos Dados

A existência de múltiplas cópias físicas da mesma informação obriga à definição de uma estratégia para a manutenção da consistência dos dados. Uma estratégia deste tipo está organizada em três vertentes: protocolo, propagação e iniciativa (ver Secção 3.6).

Em primeiro lugar, propõe-se um protocolo do tipo *actualização no primário*. Para cada registo é estabelecida uma *cópia primária*, gerida pela instituição que é responsável pela informação. As operações de leitura são feitas sobre a cópia local, enquanto que as operações de actualização apenas podem ser feitas sobre a cópia primária. A opção alternativa, de actualização global, iria aumentar em muito a complexidade do sistema. Seria necessário introduzir protocolos para a resolução de inconsistências.

Em relação à propagação, opta-se pela alternativa *assíncrona*. O sucesso da actualização na cópia primária não pode depender de actualizações de réplicas remotas. Por outro lado, o sincronismo das diversas cópias pode ser relaxado em função de uma maior flexibilidade.

Ao nível da iniciativa da actualização, opta-se pelo *envio das actualizações*. A justificação baseia-se no facto do *rácio leitura-escrita* ser alto, e, consequentemente, haver um melhor aproveitamento da rede.

Conclusões

Para a situação concreta de **partilha de informação sobre alunos entre faculdades**, propõe-se:

- Para cada registo, deve ser especificada qual a cópia primária.
- Devem ser criadas cópias físicas de cada registo nas várias instituições envolvidas na partilha.
- As operações de leitura do registo são sempre feitas sobre a cópia local, enquanto que as operações de escrita são feitas na cópia primária.
- A coerência das diversas cópias físicas é mantida através do envio de mensagens de actualização assíncronas, a partir do servidor que detém a cópia primária.

No Capítulo 6, é descrita a implementação desta arquitectura, as opções tecnológicas tomadas e as decisões ao nível da camanda de representação da informação.

5.2 Interoperabilidade Vertical

Nesta secção, descrevem-se as opções tomadas relativamente à arquitectura do sistema para os casos associados à interoperabilidade vertical. Os conceitos de *cliente* e *servidor* são utilizados com frequência ao longo desta secção. Com base em Kossmann [61], consideram-se as seguintes definições:

- **Cliente** - Sistema onde são definidas as interrogações aos dados, para consulta ou processamento. Funciona como “consumidor” dos dados. Por exemplo, no caso concreto da concentração de informação sobre recursos humanos, a Reitoria da Universidade funciona como *cliente*.
- **Servidor** - Sistema onde os dados são armazenados ou produzidos. Funciona como “fonte” dos dados. Por exemplo, no caso concreto da disseminação de notícias pelas faculdades, a Reitoria da Universidade funciona como *servidor*.

Em função da arquitectura do sistema distribuído (ver Secção 3.2), a interacção pode ser iniciada por qualquer uma das partes.

5.2.1 Agregação de Informação

Introdução

No contexto académico, cada instituição armazena e é responsável pela informação relativa às suas actividades. Pretende-se implementar um sistema distribuído que permita agregar esta informação num ponto central. Esta informação pode ser *local* - referente apenas aos dados de uma instituição; ou *global* - referente a um conjunto de dados que podem pertencer, ou não, à mesma instituição.

Este cenário constitui uma arquitectura do tipo *cliente-servidor* - um sistema (cliente) inicia a interacção enviando um pedido a outro sistema, o servidor, que envia uma resposta a este pedido. O cliente corresponde à instituição que pretende agregar a informação e os servidores correspondem às instituições que armazenam os dados. Este é um caso particular uma vez que existe apenas um cliente - o ponto onde a informação vai ser agregada. Existe uma relação de *muitos para um*, entre os dados originais e os dados agregados. Na Figura 5.1, é ilustrada esta situação.

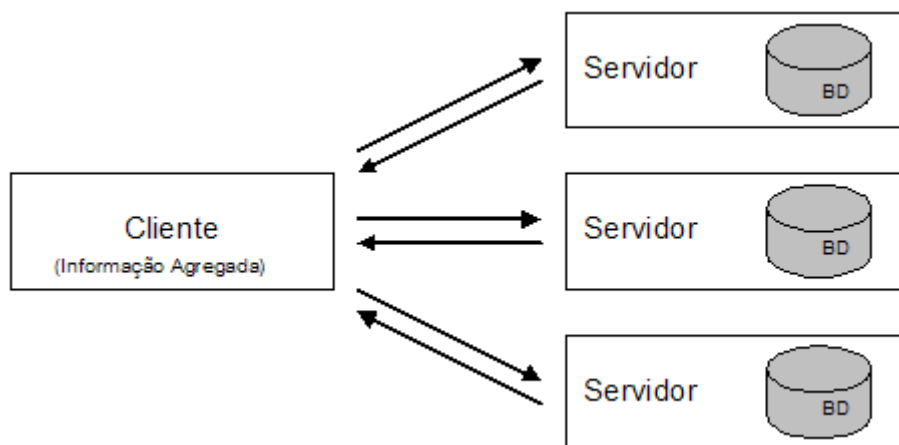


Figura 5.1: Agregação de Informação

Kossmann [61] estabelece uma distinção entre sistemas distribuídos em função das capacidades dos servidores. Duas categorias são identificadas: *sistemas homogéneos* e *sistemas heterogéneos*. Num sistema homogéneo, os diversos servidores têm funcionalidades e capacidades equivalentes (p.e. são implementados sobre um modelo de dados comum). Num sistema heterogéneo, cada servidor tem características diferentes. O nosso caso é heterogéneo, pois nem todas as instituições usam o mesmo SI.

Resumindo, este é um cenário do tipo cliente-servidor onde existe apenas um cliente e múltiplos servidores. Os servidores são heterogéneos e as interrogações podem ser locais ou globais.

Colocação dos Dados

A colocação dinâmica dos dados pode ser feita segundo três alternativas, já expostas na secção 3.5. Neste caso, a melhor opção vai depender das características das interrogações a

processar. Num primeiro nível, podemos distinguir dois tipos de interrogações, com base nos dados de que necessitam. Aquelas que só podem ser processadas movendo os dados para o cliente, e aquelas que podem ser processadas no cliente ou nos servidores.

Um exemplo de uma interrogação do primeiro tipo é o cálculo da distribuição dos alunos por percentís no âmbito da universidade. Não é possível calcular percentís parciais e depois agregar os dados. É necessário mover toda a informação para o cliente e aí fazer o processamento.

Nos casos em que as interrogações podem ser processadas no cliente ou nos servidores, há dois factores principais: custo de execução da interrogação e custo de transferência dos dados. A capacidade de processamento existente nas instituições académicas não é um critério determinante. Ao longo dos anos, tem-se assistido ao aumento da capacidade de processamento dos sistemas informáticos, associado a uma descida dos preços.

Assim, o custo determinante é o da comunicação correspondente à transferência dos dados. Testes realizados por Rodriguez-Martinez confirmam esta afirmação [84]. A colocação dos dados terá como objectivo minimizar o volume total de dados a transferir.

Rodriguez-Martinez distingue dois tipos de operadores em função do efeito que têm sobre o volume dos dados originais [84]. Refere, por um lado, *operadores de expansão* que aumentam o número ou dimensão dos tuplos no resultado (p.e. projecções ou rotações de dados) e, por outro, *operadores de redução* que reduzem os dados originais para abstrações muito menores. Assim, o critério que vai determinar a colocação dos dados é o tipo de operação realizada pela interrogação (redução ou expansão).

Por outro lado, o envio dos dados também é vantajoso em cenários onde várias interrogações são aplicadas sobre o mesmo conjunto de dados. Em vez de enviar várias vezes os mesmos dados, a manutenção de *caches* no cliente permite melhorar o desempenho global do sistema [62].

Eis dois exemplos onde o *envio dos dados* seria vantajoso:

- Imagine-se a interrogação “as últimas notícias publicadas numa instituição para cada categoria”. Cada notícia pode pertencer a múltiplas categorias. Assim, é mais “barato” enviar para o cliente a lista de todas as notícias, todas as categorias e as relações existentes e depois fazer o “desdobramento” por categorias; do que, em alternativa, fazer o processamento no servidor e enviar várias listas de notícias por cada categoria.
- Considerem-se várias interrogações a serem aplicadas a uma lista de alunos de um curso: “lista de alunos ordenada alfabeticamente”, “lista de alunos ordenada por média final”, “lista de alunos ordenada por inscrições totais”, etc. Neste caso, é evidentemente mais vantajoso mover os dados para a *cache* do cliente e aí fazer as várias interrogações.

Concluindo, deve ser adoptada uma estratégia do tipo *envio da interrogação* quando a execução da interrogação produz uma *redução* do volume de dados original. Alternativamente, deve ser adoptada uma estratégia do tipo *envio dos dados* quando a interrogação provoca um *aumento* no volume de dados; ou quando várias interrogações são executadas sobre os mesmos dados (p.e. análise de dados).

O sistema MOCHA [84] adopta uma solução híbrida que, com base num parâmetro designado por *Factor de Redução de Volume*, selecciona um dos dois mecanismos.

Interfaces de Comunicação

Em relação à capacidade dos sistemas, e como já foi referido, este cenário é heterogéneo. Os sistemas de informação das faculdades são independentes e as características de cada implementação, desconhecidas. Para estes casos, a tendência geral é a de tentar reduzir este problema a um de sistemas homogéneos através da implementação de interfaces de acesso comuns [61]. A homogeneidade é conseguida, impondo um modelo de dados global sobre o modelo de dados local usado por cada servidor [84].

Assim, os servidores devem implementar interfaces comuns que permitam ao cliente interagir da mesma forma com todos eles.

Sincronismo

Este caso constitui um sistema clássico do tipo *pedido-resposta*: pedidos aperiódicos num padrão 1-1 (cada pedido gera uma resposta) [42]. A implementação de um sistema deste tipo pode seguir uma de duas abordagens: *síncrona* ou *assíncrona* (ver secção ref-sec:Sincronismo). No primeiro cenário, o cliente faz o pedido e bloqueia à espera da(s) resposta(s). No caso assíncrono, o cliente emite o pedido e não fica bloqueado; num momento posterior, o servidor envia a resposta.

Na agregação de informação, opta-se por um sistema com pedidos síncronos. A principal justificação para esta opção é a maior facilidade de implementação. Apesar de resultar numa arquitectura mais acoplada do ponto de vista temporal, a *dependência tecnológica* é menor.

Conclusões

Resumindo, para o cenário de agregação de informação entre instituições académicas, propõe-se:

- Uma arquitectura do tipo *envio da interrogação*, quando as operações a efectuar reduzem o volume de dados inicial. Isto é, casos onde o volume de dados iniciais é maior do que o volume de resultados.
- Uma arquitectura do tipo *envio dos dados* em três casos: quando as interrogações só podem ser processadas desta forma, quando aumentam o volume de dados ou quando fazem parte de um conjunto de interrogações que vão ser aplicadas sobre os mesmos dados.
- A *especificação de interfaces procedimentais* (API) para a implementação de um mecanismo de processamento distribuído homogéneo entre os servidores.
- Uso de um *mecanismo síncrono* para a implementação dos pedidos e respostas.

Caso de Estudo

O caso de estudo - **agregação de informação estatística sobre cursos** - é um caso particular da agregação de informação. Assim, as várias opções expostas na secção anterior mantêm-se. Em relação à colocação dos dados, opta-se pelo processamento da interrogação no servidor (*envio da interrogação*), porque o cálculo de estatísticas envolve operações que têm por objectivo produzir abstrações muito menores dos dados originais - operação de redução. Com esta opção, os custos de comunicação são menores, relativamente à alternativa de envio dos dados.

5.2.2 Concentração de Informação

Introdução

Com a concentração de informação pretende-se disponibilizar, de forma integrada e num único sistema, toda a informação existente sobre um determinado domínio. Esta informação existe de forma distribuída, dispersa por várias instituições.

Neste caso, existem vários servidores - as instituições que detêm a informação - e apenas um cliente, que corresponde ao ponto de integração. Aqui, e ao contrário da situação anterior, o mapeamento da informação é de *um-para-um*. Para cada item existente nos servidores, existe um acesso no sistema que faz a integração.

Colocação dos Dados

Relativamente à estratégia de colocação dos dados, neste caso não existe nenhum *factor de redução* que justifique uma abordagem do tipo *envio da interrogação*. Há uma correspondência directa entre os dados nos servidores e os dados no cliente. Por outro lado, sobre os mesmos dados podem, em função do caso de estudo, ser realizadas várias interrogações.

Assim, para evitar latências e reduzir os custos de comunicação, opta-se por *manter cópias da informação no cliente*. Para cada dado lógico, podem existir duas representações físicas.

Consistência dos Dados

Como existem várias cópias físicas da mesma informação, é necessário definir uma estratégia para a manutenção da consistência dos dados. Esta estratégia envolve três dimensões: protocolo, propagação e iniciativa (ver Secção 3.6).

Em relação ao **protocolo**, propõe-se uma abordagem do tipo *actualização no primário*. Do ponto de vista do cliente, o acesso para leitura é sempre feito sobre a cópia local, enquanto que o acesso para escrita é sobre o servidor - na cópia primária. Esta opção reduz a complexidade do sistema e optimiza os custos de transferência de informação.

Ao nível da **propagação**, opta-se pela alternativa *assíncrona*. Desta forma, a arquitectura do sistema é mais tolerante a falhas e desacoplada. As actualizações locais, feitas nos servidores, não podem depender da disponibilidade da rede e do sistema cliente para terem sucesso. Após a actualização da cópia primária, e numa transacção independente, o cliente é actualizado.

Em relação à **iniciativa** da actualização, a opção vai depender do *rácio leitura-escrita* da situação em concreto. Um valor elevado, corresponde a um cenário onde o número de operações de leitura sobre a informação é muito superior ao número de operações de actualização. Um valor baixo, corresponde a uma situação onde as operações de actualização são frequentes em relação às operações de leitura.

Assim, para cenários com um *rácio elevado*, propõe-se uma estratégia do tipo envio das actualizações. Alternativamente, em situações onde o *rácio é pequeno*, propõe-se o pedido das actualizações pelo cliente.

Estas opções minimizam os custos de comunicação. Quando as operações de leitura são muitas, evitam-se os pedidos de informação que não foi alterada. Neste caso, o servidor envia as actualizações. Quando são mais as operações de escrita, evita-se a “avalanche” de mensagens e interrupções associadas ao envio das muitas actualizações. Aqui, o cliente pede as actualizações.

Em relação à opção entre envio das actualizações ou envio de notificações, opta-se pela primeira hipótese porque o número de mensagens trocadas é menor. O envio de notificações é uma opção mais interessante quando o volume dados em cada mensagem é elevado, e as actualizações são muito frequentes.

Conclusões

Resumindo e sistematizando, para o caso da **concentração da informação**, propõe-se:

- A manutenção de *cópias físicas da informação no cliente (primária) e nos servidores (réplica)*.
- O uso de um protocolo do tipo *actualização no primário*, com propagações *assíncronas*, para a gestão da consistência da informação.
- Uma iniciativa do tipo *envio das actualizações*, quando o *rácio leitura-escrita* é alto.
- Uma iniciativa do tipo *pedido dos dados*, quando o *rácio leitura-escrita* é pequeno.

Caso de Estudo

Em relação ao caso de estudo - **concentração de informação sobre recursos humanos**, o objectivo é viabilizar a centralização, na Reitoria da Universidade (*cliente*), de dados referentes aos recursos humanos associados às diversas instituições que integram a Universidade (*servidores*). A informação está estruturada pelas diversas instituições. Cada instituição armazena e é responsável por parte significativa da informação relativa à actividade dos seus recursos humanos.

Sendo este um caso particular da concentração de informação, adoptam-se as opções expostas nas subsecções anteriores. No ponto particular da iniciativa da propagação, o *rácio leitura-escrita* é relativamente elevado. As consultas sobre informação relativa aos recursos humanos é mais frequente do que as alterações feitas a estes dados. Assim, propõe-se o *envio das actualizações*.

5.2.3 Difusão de Informação

Introdução

Neste caso, a informação existe num único servidor e pretende-se disseminá-la por vários clientes. É um cenário onde um item é “multiplicado” e distribuído por vários clientes - padrão do tipo *um para muitos*. A informação é produzida pelo servidor e os clientes apenas podem realizar operações de leitura.

Neste contexto, distinguimos dois tipos de difusão de informação: *difusão controlada* e *difusão pública*. Nas subsecções seguintes, apresentam-se estes dois cenários e as arquitecturas propostas.

Difusão Controlada

Neste caso, os clientes constituem uma população limitada e bem conhecida à priori. O servidor conhece os clientes e as necessidades de cada um. Pode estabelecer, internamente, um mapeamento entre a informação e cada um dos clientes. Por outro lado, com uma difusão controlada, pretende-se garantir a entrega da informação.

As alternativas relativas à **entrega dos dados** foram expostas na Secção 3.7. Para a difusão controlada de informação, opta-se por uma abordagem do tipo *envio da informação de forma aperiódica* - a interacção é iniciada pelo servidor e a informação é enviada quando surge.

Esta opção é justificada pelos seguintes argumentos:

- É possível garantir a consistência da informação entre o servidor e os clientes. Independentemente da periodicidade das actualizações, não há informação perdida.
- A utilização da rede é optimizada. Todas as mensagens que circulam são relevantes e contêm informação nova.
- O número de clientes é reduzido e conhecido à partida. Por isso, o esforço computacional e de armazenamento de informação (estado dos clientes) é previsível e controlável.

Em relação ao **sincronismo da interacção**, opta-se por uma *solução assíncrona* devido à maior tolerância a falhas, à melhor adequação a cenários “um para muitos” e à maior escalabilidade [76]. Num cenário síncrono, a falha de um cliente na recepção da informação é uma situação crítica, que tem impacto sobre todo o sistema.

Difusão Pública

Neste cenário, pretende-se disseminar a informação por uma população interessada em receber a informação, mas desconhecida à priori. Não são conhecidos os interesses nem o número de clientes. A entrega garantida não é um requisito, o acesso à informação depende do cliente. Como a informação é pública, não é necessário controlo de acessos.

Para a **entrega dos dados**, propõe-se uma arquitectura do tipo *pedido da informação de forma aperiódica* - a interacção é iniciada por cada cliente em qualquer momento.

Esta arquitectura justifica-se porque:

- A consistência da informação não é um requisito - o ónus é deslocado para o cliente.
- A escalabilidade da solução é fundamental. O número de clientes é desconhecido e com “vontades” heterogéneas. Com esta opção, o servidor não necessita de manter informação de estado, nem de processar o envio de cada mensagem.

Em relação ao **sincronismo da interacção**, opta-se por uma *solução síncrona*. Com a alternativa assíncrona, o servidor seria obrigado a manter informação sobre cada pedido para poder enviar a resposta. Desta forma, a escalabilidade do sistema seria prejudicada.

Casos de Estudo

No caso da difusão de informação, existem dois casos de estudo: *difusão controlada de notícias pelas faculdades* e *difusão pública de notícias* (Figura 5.2).

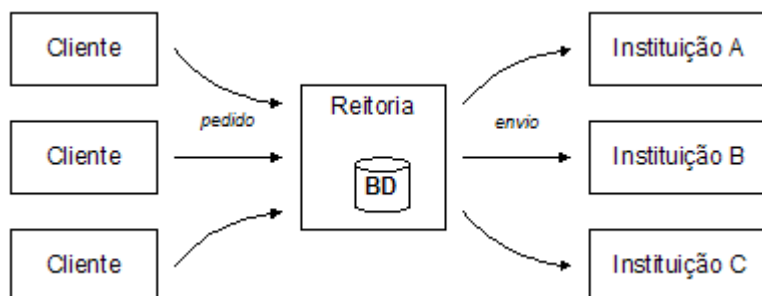


Figura 5.2: Difusão de Informação - Instituições da Universidade e Público em Geral

No caso da **difusão controlada de notícias pelas faculdades**, pretende-se disponibilizar notícias, a partir da Reitoria (*servidor*), pelas instituições da Universidade (*clientes*). O mapeamento entre as notícias e as instituições é feito na Reitoria.

A solução proposta - envio da informação de forma aperiódica e assíncrona - permite controlar a distribuição das notícias em função da instituição de destino. É possível, por exemplo, enviar um item apenas para uma ou um conjunto de instituições. No entanto, a utilização de granularidades mais finas, como enviar, por exemplo, um item para uma secção dentro de uma instituição, torna-se impossível. Para tal, seria necessário que a Reitoria mantivesse informação sobre a estrutura interna de cada instituição. Uma abordagem possível seria a definição e utilização de meta-informação, recorrendo, por exemplo, a ontologias. Esta opção ultrapassa largamente o âmbito desta dissertação e não será, por isso, explorada.

No outro caso de estudo - **difusão pública de informação noticiosa** - pretende-se permitir “à comunidade em geral” ter acesso, de forma programática, a um subconjunto das notícias produzidas.

Neste caso, o servidor publica a informação que pretende colocar ao dispor da comunidade. Os clientes pedem ao servidor, em qualquer momento, a informação que pretendem. A informação pode ser disponibilizados de forma permanente ou durante um período de tempo específico, dependendo da sua natureza. Para o caso das notícias, propõe-se um intervalo mínimo mensal.

Com o objectivo de facilitar a interoperabilidade e minimizar o impacto sobre os clientes, propõe-se o uso de especificações públicas para a representação da informação. Assim, os clientes não necessitam de desenvolver interfaces específicas para aceder à informação publicada.

Capítulo 6

Implementação

Neste capítulo, é apresentada a implementação dos casos de estudo seleccionados. Depois de uma breve referência às opções e convenções técnicas, o capítulo está estruturado em função dos casos de estudo: partilha do registo académico do aluno (6.3), agregação de estatísticas sobre cursos (6.4), concentração de informação sobre os recursos humanos (6.5) e difusão de notícias da universidade (6.6). No final do capítulo (6.7), é feita uma breve referência aos protótipos implementados.

6.1 Opções Tecnológicas

6.1.1 XML Schema

Para a representação dos dados, optou-se por usar a linguagem *XML Schema* [38]. A linguagem XML é actualmente a norma *de facto* para a representação de informação estruturada [67]. Face à alternativa - *Document Type Definition* (DTD), a opção é justificada com base nos seguintes argumentos:

- O XML Schema permite usar *namespaces*, o que possibilita uma melhor estruturação dos documentos.
- O XML Schema oferece uma gama de tipos de dados base mais abrangente do que a que é oferecida pelos DTD. Com XML Schema é ainda possível definir tipos próprios.
- O XML Schema permite um maior controlo na validação dos documentos.
- O XML Schema tem um mecanismo de reutilização de dados, baseado em conceitos orientados a objectos (OO), mais sofisticado do que o que é oferecido pelos DTD.
- Os documentos em XML Schema são documentos XML “bem formados”, enquanto que os DTD são expressos usando outra sintaxe (EBNF). Quando se usam DTD, é necessário dominar duas linguagens distintas.
- O XML Schema é usado e suportado por várias organizações.

A mesma opção foi tomada no contexto de outras iniciativas, como é o caso no PESC [22].

Existem, por outro lado, muitas especificações que se integram com a linguagem XML e permitem adicionar novas “funcionalidades”. As especificações XML-Signature [14] e XML-Encryption [13] são um exemplo. A primeira permite associar assinaturas digitais a qualquer conteúdo em XML. A segunda permite cifrar dados (desde documentos XML a elementos individuais) e representar o resultado em XML. Estes mecanismos permitem cumprir os requisitos de *autenticidade*, *integridade* e *confidencialidade* estabelecidos no Capítulo 4.

Concluindo, opta-se pelo XML Schema para representar a informação trocada entre as instituições. Em concreto, usam-se documentos XML para representar: o registo académico do aluno, as estatísticas sobre os cursos, a informação sobre recursos humanos e os itens noticiosos.

6.1.2 Serviços Web

Para implementação dos mecanismos de interacção entre os sistemas distribuídos, opta-se pelo uso de Serviços Web (Secção 3.8.3). Os Serviços Web oferecem características únicas [8] e vantagens bem conhecidas [67] como: a independência ao nível da plataforma tecnológica, simplicidade de implementação, desacoplamento entre as partes e um forte

apoio da indústria. Actualmente, são várias as tecnologias associadas ao conceito de Serviço Web; seleccionando-se duas em particular: URI e HTTP.

Os *Uniform Resource Identifiers* (URI) [5] permitem identificar os recursos disponibilizados e os pontos de acesso às interfaces publicadas por cada um dos sistemas. São caracterizados por três definições: *uniformidade*, o que permite uma interpretação semântica comum; *recurso*, um modelo conceptual de uma entidade física ou lógica; *identificador*, que funciona como uma referência para algo que tenha uma identidade.

O *HyperText Transfer Protocol* (HTTP) [39] é um protocolo genérico que, para além do uso tradicional associado ao hipertexto, pode ser utilizado em diversas tarefas, como na gestão de recursos distribuídos. A opção pelo HTTP permite estabelecer uma interface de acesso comum, com recurso aos métodos definidos: GET, POST, PUT e DELETE. A norma também especifica um conjunto de códigos de erro que podem ser usados no contexto das interacções, como por exemplo: 200 OK, 201 Created, 403 Forbidden, 404 Not Found.

Uma outra vantagem associada à opção pelo protocolo HTTP é a existência de especificações que permitem implementar mecanismos de autenticação para controlo de acessos [45] e canais de comunicação seguros, recorrendo a normas como o HTTPS [80]. Desta forma, é possível implementar alguns dos requisitos de segurança estabelecidos no Capítulo 4.

Na estratégia seguida para a implementação dos Serviços Web, procurou-se maximizar a integração com as infraestruturas Web já existentes (*caches*, *proxies*). Uma das consequências desta opção é a maior interoperabilidade com os sistemas existentes. Faz-se uma última referência às opções tomadas no desenho da arquitectura da WWW [40], que foram um contributo importante para o trabalho aqui apresentado.

6.2 Convenções Técnicas

Nesta secção, são apresentadas as convenções técnicas que devem ser seguidas na elaboração da implementação final. O desenvolvimento deste conjunto de recomendações teve por base 3 documentos: XML Forum Technical Specification for Higher Education [23], ASC X12 Reference Model for XML Design [12] e XML Schemas - Best Practices [15].

6.2.1 Idioma

Deve ser usado o idioma Inglês para representar os termos usados na codificação e comunicação. Por exemplo: “<Student>” em vez de “<Estudante>”. O carácter internacional do projecto e, em concreto, a troca de registos académicos no âmbito do programa ERASMUS justificam esta opção.

6.2.2 Nomenclaturas

Deve ser usada a norma designada por *Upper Camel Case* (UCC) na escrita dos termos. Esta regra determina a junção dos nomes num termo (eliminando os espaços) e o uso de maiúsculas para cada palavra (Por exemplo: “StudentHomeAddress”). Esta regra tem

sido adoptada noutros casos [23] e constitui uma prática comum em documentos deste tipo. A principal justificação relaciona-se com a facilidade de leitura resultante.

Não devem ser usados os seguintes caracteres nos termos: “_” (*underscore*), “-” (hífen) e “.” (ponto).

Quando se definem tipos, deve ser acrescentado o sufixo “Type” ao termo (Por exemplo: “PersonType”).

6.2.3 Codificação

Para a codificação dos documentos deve ser usado o formato *Unicode Transformation Format* de 8 bits (UTF-8). A Internet Engineering Task Force (IETF) recomenda [1] o uso deste formato.

6.2.4 Uso de Elementos ou Atributos

Este é um tema de debate frequente - “Elementos vs. Atributos”. A opção por uma das duas formas de estruturar a informação não deve ser absoluta. Neste caso, vamos adoptar as regras usadas pelo XML Forum [23]. O método proposto define 4 passos para ajudar a determinar o recurso a elementos ou atributos:

- Determinar se os dados em questão são fundamentalmente conteúdo ou metadados. Usar elementos no primeiro caso.
- Determinar os requisitos estruturais para os dados. Usar elementos quando é necessária a validação de estruturas complexas.
- Determinar como vão ser usados os dados: para transmitir informação sobre o domínio ou para o processamento de informação. Usar elementos no primeiro caso.
- Usar atributos para reforçar a relação de um para um entre partes da informação. Para, por exemplo, representar tuplos de informação.

Nestes casos de estudo, um dos principais objectivos é a transferência de informação. Por outro lado, os documentos definidos têm uma estrutura complexa. Assim, na generalidade das situações, serão usados elementos. Uma vantagem que advém desta opção é a maior facilidade de leitura dos documentos XML.

6.3 Partilha do Registo Académico do Aluno

Nesta secção, apresenta-se o trabalho desenvolvido na implementação do Registo Académico do Aluno, cumprindo os requisitos estabelecidos antes.

6.3.1 Identificador do Registo Académico do Aluno

Para a implementação de um identificador único para o registo académico do aluno optou-se por uma estrutura hierárquica. Cada aluno é identificado de forma única, compondo o código de aluno, o código da instituição e o código do país.

Código do País - Código da Instituição - Código do Aluno

O código identificador do país segue a norma ISO 3166 [41], que define um código numérico de 3 dígitos para cada país. Por exemplo:

- **620** - Portugal
- **276** - Alemanha
- **724** - Espanha

O segundo nível, que corresponde à identificação das instituições de ensino superior, é da responsabilidade do país. Para cada país, deve ser definida uma entidade responsável pela atribuição de identificadores às instituições de ensino superior. Em Espanha, e no âmbito do projecto SCANet, foi escolhida a AECOC, a entidade responsável pelos códigos EAN em Espanha, para a atribuição de códigos às instituições.

Em Portugal, o Ministério da Ciência e do Ensino Superior (MCES) define códigos para os estabelecimentos de ensino e cursos [24]. Por exemplo:

- **1105** - Faculdade de Engenharia da Universidade do Porto
- **0502** - Faculdade de Direito da Universidade de Coimbra

O último nível do código corresponde ao identificador interno mantido pela faculdade para cada aluno. Assim, o código **620-1105-010570016** corresponde a:

- **620** - Portugal
- **1105** - Faculdade de Engenharia da Universidade do Porto
- **010570016** - Sérgio Sobral Nunes (aluno do MGI)

6.3.2 Representação do Registo Académico do Aluno

O esquema proposto pelo PESC, para a representação de informação sobre estudantes [19], foi usado como ponto de partida para o trabalho aqui apresentado. Esta norma destina-se prioritariamente à partilha de dados e é dirigida a um número significativo de instituições. Assim, procurou-se destacar a semântica da informação e eliminar ambiguidades ao nível dos tipos de dados usados.

A raiz da especificação é o elemento `StudentRecord`, que representa um Registo Académico de Aluno (Figura 6.1). Este elemento é composto por dois subelementos obrigatórios: `DocumentInfo` e `Student`. O elemento `Note`, que pode não existir ou existir várias vezes, representa uma anotação.

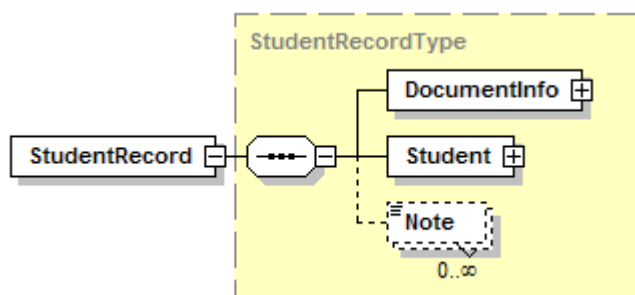


Figura 6.1: Elemento StudentRecord

O elemento `DocumentInfo` é do tipo `DocumentInfoType` e reúne informação sobre o documento (Figura 6.2). Este elemento é composto por quatro subelementos obrigatórios:

- `DocumentCode` - Referência do documento. O valor atribuído é da responsabilidade da entidade que cria o documento.
- `DocumentIssueDate` - Data de criação deste documento.
- `Issuer` - Informação sobre a entidade responsável pela emissão deste documento.
- `StudentCode` - Número do aluno definido pela entidade emissora.

Reunindo a informação da instituição (`Issuer`) e o identificador interno de cada aluno (`StudentCode`), é possível identificar de forma universal todos os aluno.

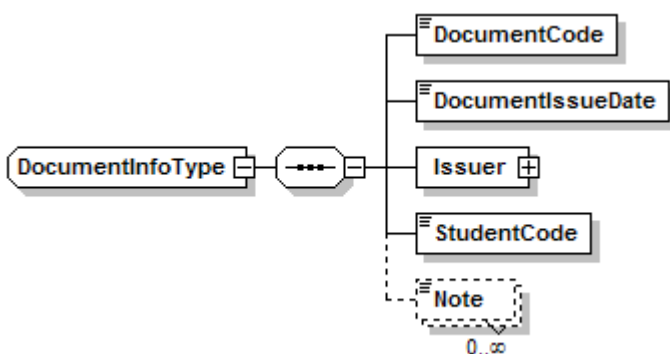


Figura 6.2: Elemento DocumentInfoType

O elemento `Student` é do tipo `StudentType` (Figura 6.3), que define a estrutura para a informação relativa a um estudante e inclui:

- `Person` - Reúne informação sobre a pessoa; é um elemento do tipo `PersonType`.
- `Family` - Informação sobre a filiação do aluno. Contém dois subelementos (`Mother` e `Father`) do tipo `PersonType`.

- **AdmissionExam** - Representa uma prova de admissão efectuada para candidatura ao ensino superior. Um aluno pode conter várias provas de admissão. Sobre cada prova são armazenados o título, o código, a data e o resultado.
- **AcademicDegree** - Representa um grau académico que o aluno possui ou no qual está matriculado. Um aluno pode ter vários graus académicos. Este subelemento é do tipo **AcademicDegreeType**.
- **HealthRecord** - Registo de saúde do aluno. Inclui informação sobre exames e imunizações do aluno.
- **MilitaryRecord** - Informação sobre a situação militar do aluno, em concreto qual o estado desse processo.

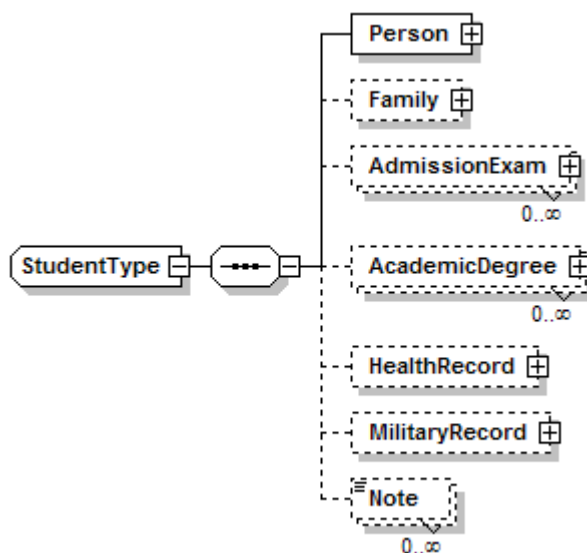


Figura 6.3: Elemento StudentType

O elemento **PersonType** (Figura 6.4) representa a informação base sobre uma pessoa; é usado para representar informação sobre o aluno e respectiva filiação. A estrutura deste elemento inclui:

- **IdentificationCode**: Representa um elemento de identificação: código, descrição, valor, emissor, data de emissão e data de validade. Sobre uma pessoa é possível reunir vários subelementos deste tipo.
- **Name**: Representa o nome da pessoa, estruturado em título, primeiro nome, nome(s) do meio(s), último nome e nome completo.
- **Birth**: Informação sobre o nascimento. Inclui data de nascimento, cidade e país.
- **Deceased**: Informação sobre o falecimento. Este subelemento existe no caso de falecimento do aluno e inclui indicador de falecimento e data.

- Gender: Informação sobre o género do aluno.
- MaritalStatus: Estado civil do aluno.
- Citizenship: Informação sobre o país de origem do aluno.
- Residency: Informação sobre a residência do aluno: morada, código postal, cidade e país.
- Contact: Conjunto de contactos do aluno. Cada contacto pode ser do tipo: endereço postal, telefone, endereço de correio electrónico ou página Web.

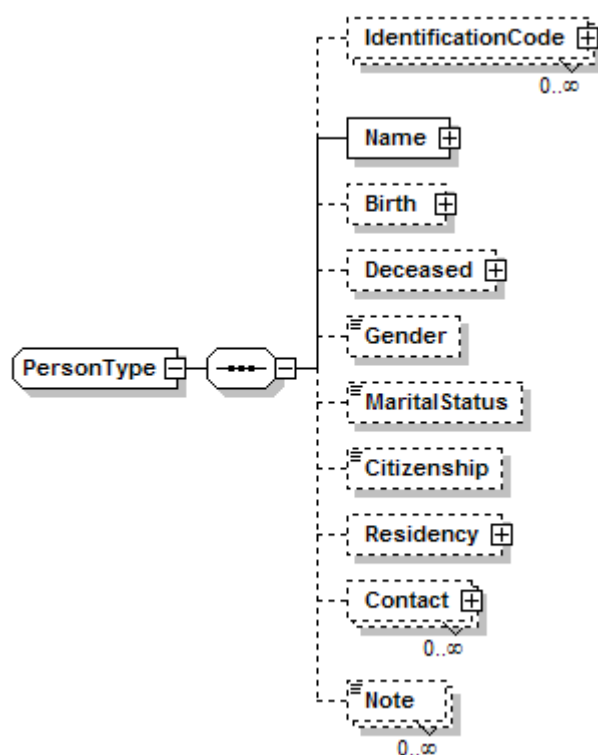


Figura 6.4: Elemento PersonType

O elemento AcademicDegreeType (Figura 6.5) representa uma matrícula num grau académico e inclui:

- Degree: Informação sobre o grau académico, do tipo DegreeType.
- StudentCode: Código do aluno na instituição responsável pelo grau.
- DegreeAdmissionDate: Data de matrícula no grau académico.
- DegreeStudentStatus: Estado da matrícula no grau académico (a frequentar, permutado, concluído, etc).

- `DegreeConclusionDate`: Data de conclusão do grau académico. A não existência deste subelemento significa que o grau não foi terminado.
- `DegreeAcademicGradeAverage`: No caso do curso estar concluído, representa a média final do aluno na escala usada na instituição.
- `DegreeECTSGradeAverage`: A média final do aluno na escala ECTS.
- `AcademicSession`: Representa uma sessão académica do tipo `AcademicSessionType`. Cada sessão académica corresponde a um período lectivo.
- `AcademicAward` - Informação sobre prémios académicos atribuídos ao aluno no contexto de um grau académico. Em relação a cada prémio (do tipo `AcademicAwardType`), é armazenado o título e a data.
- `ExtraActivity` - Informação sobre actividades extra-curriculares do aluno. Um aluno pode ter várias actividades extra-curriculares no contexto de um grau académico. Sobre cada actividade é registado o nome, a data de início, a data de conclusão e a descrição.

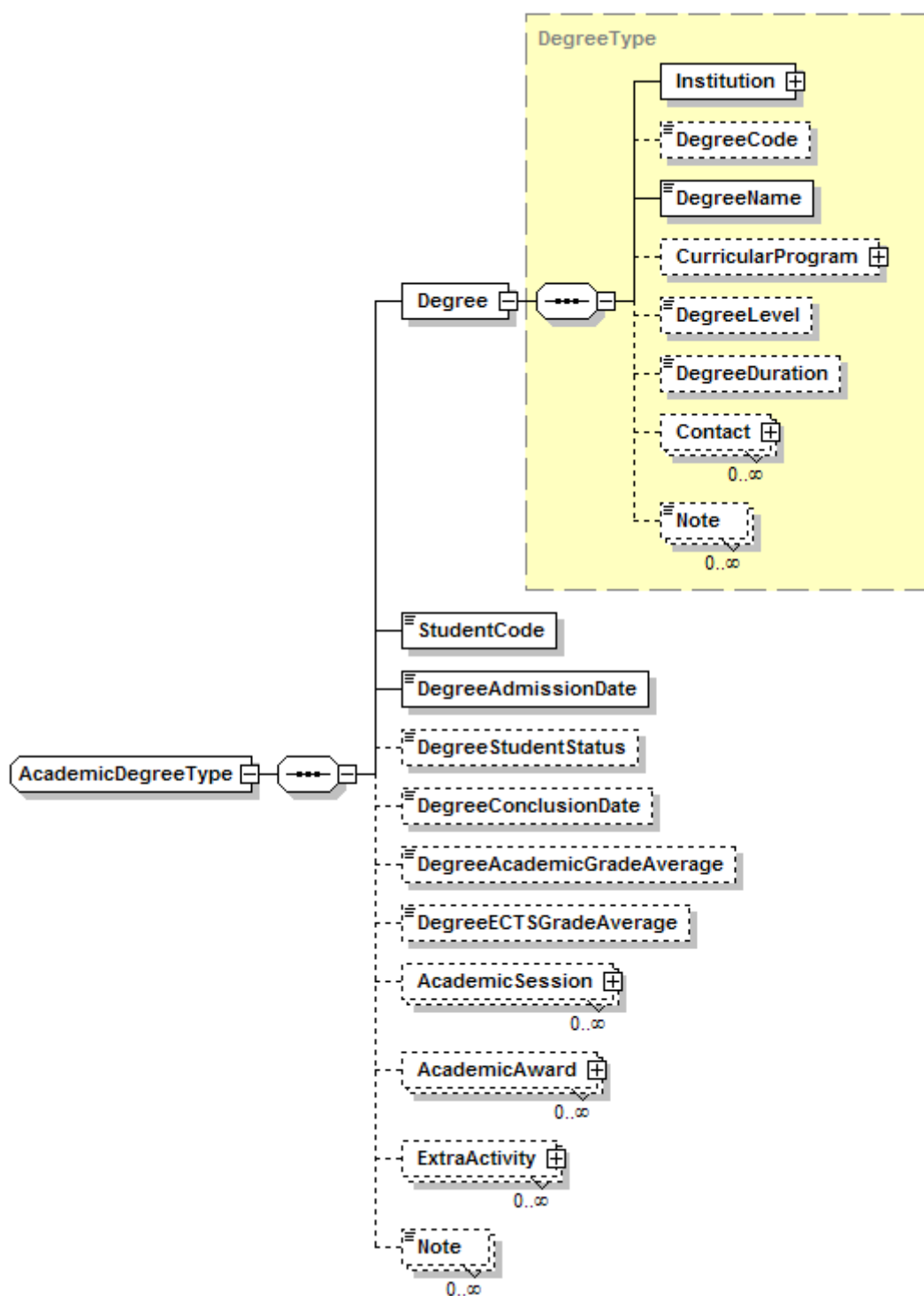


Figura 6.5: Elementos AcademicDegreeType e DegreeType

O **DegreeType** (Figura 6.5) reúne informação sobre o grau académico:

- **Institution**: Informação sobre a instituição responsável pelo curso.
- **DegreeCode**: Código do grau académico, atribuído pela instituição.

- DegreeName: Nome do grau académico.
- CurricularProgram: Referência para o plano de estudos. O código de cada plano é definido pela instituição.
- DegreeLevel: Nível académico deste grau; os valores possíveis são bacharelato, licenciatura, pós-graduação, mestrado e doutoramento.
- DegreeDuration: Duração oficial deste grau académico.
- Contact: Conjunto de contactos do curso: morada, página Web, endereço de correio electrónico, etc.

Como já foi referido, cada período lectivo associado à frequência de um grau académico é representado pelo elemento `AcademicSessionType` (Figura 6.6).

Este elemento contém:

- `AcademicSessionStartDate`: Data de início do período lectivo.
- `AcademicSessionEndDate`: Data de conclusão do período lectivo.
- `Course`: Conjunto de inscrições em disciplinas feitas durante o período lectivo. Cada disciplina é do tipo `CourseType`.
- `Thesis`: Informação sobre uma eventual dissertação realizada no âmbito deste período lectivo e que inclui o título da dissertação, nome do(s) supervisor(es) e avaliação.

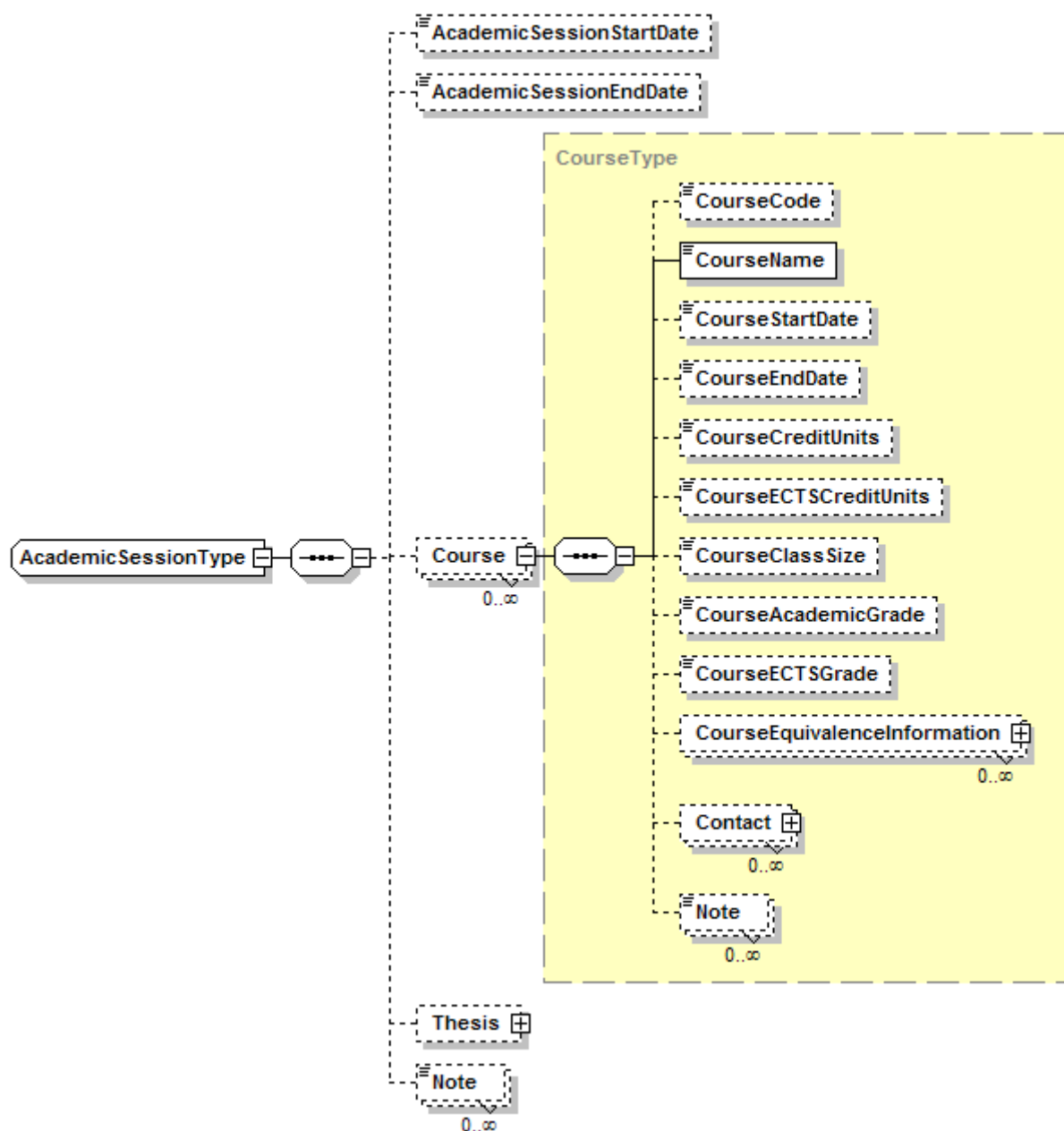


Figura 6.6: Elementos `AcademicSessionType` e `CourseType`

Cada disciplina é representada pelo elemento `CourseType` (Figura 6.6) e inclui:

- `CourseCode`: Código da disciplina, atribuído pela instituição.
- `CourseName`: Nome da disciplina.
- `CourseStartDate`: Data de início.
- `CourseEndDate`: Data de conclusão.
- `CourseCreditUnits`: Unidades de crédito associadas à disciplina.

- `CourseECTSCreditUnits`: Unidades de crédito ECTS.
- `CourseClassSize`: Número de alunos inscritos.
- `CourseAcademicGrade`: Classificação do aluno na disciplina. O valor é expresso na escala usada na instituição.
- `CourseECTSGrade`: Classificação do aluno na escala ECTS.
- `CourseEquivalenceInformation`: No caso da aprovação a esta disciplina ter sido obtida através de uma equivalência, este subelemento permite armazenar essa informação. É do tipo `EquivalenceInformationType`.
- `Contact`: Conjunto de contactos associados à disciplina, por exemplo: endereço de correio electrónico, página Web.

Nesta apresentação dos principais elementos que compõem a especificação proposta, foi omitida a descrição repetida do subelemento `Note` existente em vários elementos. Este elemento é do tipo `NoteType` e representa uma anotação que é possível associar aos elementos.

A listagem completa da especificação do Registo Académico do Aluno, em XML Schema, encontra-se no Apêndice B.

6.3.3 Especificação do Sistema

Diagrama de Componentes

Na Figura 6.7, apresenta-se o diagrama de componentes para o cenário de partilha de informação entre instituições. O nó `Institution (Client)` representa uma instituição que detém réplicas da informação. O nó `Institution (Server)` representa uma instituição que detém a informação a partilhar (*primário*). Como já foi referido antes, podem existir vários servidores e vários clientes.

No caso de uma instituição funcionar como servidora de parte da informação e como cliente de outra, os dois componentes (`ClientManager` e `ServerManager`) seriam implementados no mesmo nó.

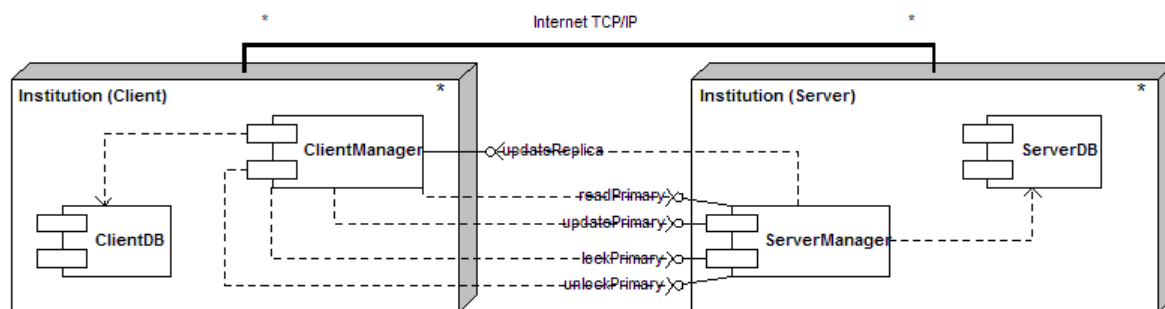


Figura 6.7: Diagrama de Componentes para a Partilha de Informação sobre Alunos

Especificação das interfaces:

- **updateReplica** - Recebe, via HTTP/POST, o identificador do aluno (`StudentRecordId`) e o registo académico em XML (`StudentRecord`). Se o pedido é válido, actualiza a réplica local.
- **readPrimary** - Recebe, via HTTP/GET, o identificador do aluno. Se o pedido é válido, responde com o documento XML correspondente ao registo académico.
- **updatePrimary** - Recebe, via HTTP/POST, o identificador do aluno (`StudentRecordId`), a chave associada ao *lock* (`LockKey`) e o registo académico em XML (`StudentRecord`). Se o pedido é válido, é actualizada a cópia primária e iniciado o processo de actualização das réplicas. Em caso de insucesso, é enviado um código de erro.
- **lockPrimary** - Recebe, via HTTP/GET, o identificador do aluno. É efectuado um *lock* sobre o recurso. Se o procedimento é bem sucedido, é enviado na resposta um código associado ao *lock* criado. Se não é possível criar o *lock*, é enviado um código de erro.
- **unlockPrimary** - Recebe, via HTTP/GET, o identificador do aluno e o código associado ao *lock* (`LockKey`). Se o pedido é válido, é libertado o *lock* sobre o registo.

Diagrama de Sequência

Na Figura 6.8, é ilustrado o processo de actualização de informação a partir do servidor. Depois de concluída a actualização da informação no servidor, é lançado o processo de actualização das várias réplicas dispersas pelos clientes. Este processo (`updateReplicas`), vai invocar a interface `updateReplica` para cada uma das réplicas e garantir que a actualização seja concluída com sucesso.

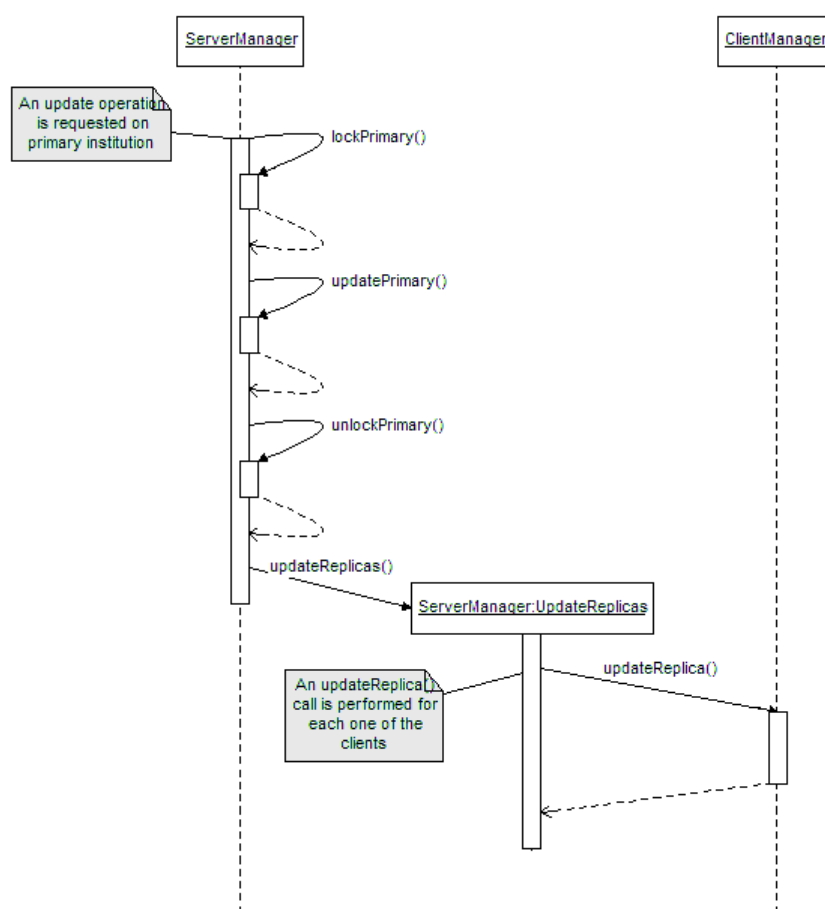


Figura 6.8: Diagrama de Componentes para a Partilha de Informação sobre Alunos

Quando o processo de actualização é iniciado pelo cliente, temos um diagrama de sequência semelhante ao que foi apresentado para o caso da concentração de informação (Figura 6.15).

6.4 Agregação de Estatísticas sobre os Cursos

6.4.1 Identificador do Curso

Para a atribuição de identificadores únicos aos cursos, foi seguida a mesma estratégia adoptada a propósito do registo académico do aluno (ver Secção 6.3.1). Assim, propõe-se um código hierárquico com a seguinte estrutura:

Código do País - Código da Instituição - Código do Curso

O código do país corresponde à norma ISO 3166; o código da instituição, e no caso de Portugal, é atribuído pelo MCES; o código do curso é atribuído pela própria instituição. Desta forma, é possível identificar universalmente cada curso.

6.4.2 Representação de Estatísticas sobre o Curso

O elemento `DegreeStatistics` é a raiz da especificação e agrega um conjunto de dados de índole estatística relativas a um curso académico (Figura 6.9). Na selecção dos dados, foram usados como ponto de partida, os relatórios DIMAS solicitados pelo MCES. Os primeiros cinco elementos permitem identificar o documento e o curso e delimitar o período a que se referem as estatísticas:

- `DocumentInfo` - Informação sobre o documento e que inclui: código interno, data de emissão e emissor.
- `DegreeIdentification` - Identificação do curso, composta por três subelementos: código do país, código da instituição e código interno do curso.
- `DegreeAcronym` - Sigla do curso.
- `DegreeName` - Nome do curso.
- `DegreeStatisticsStartDate` - Data que delimita o início do período a que se reportam as estatísticas.
- `DegreeStatisticsEndDate` - Data que delimita o fim do período a que se reportam os estatísticas.

Os restantes subelementos são optativos e reúnem a informação estatística que se considerou importante. Os valores associados às classificações são apresentados na escala usada na instituição. Sempre que se apresentam totais relativos a pessoas, é usado o tipo `PersonsNumberType`. Este tipo permite estruturar a informação em pessoas do sexo masculino, feminino, e somatório total.

- `DegreeFaculty` - Número de docentes associados ao curso.
- `AdmissionVacancies` - Número de vagas existentes para acesso ao curso (*Numeros Clausus*).
- `AcceptedApplicantsNumber` - Número de candidatos admitidos.
- `AcceptedApplicantsGradeAverage` - Média aritmética das classificações dos candidatos admitidos.
- `AcceptedApplicantsGradeMedian` - Mediana das classificações dos candidatos admitidos.
- `AcceptedApplicantsGradeStandardDeviation` - Desvio padrão das classificações dos candidatos admitidos.
- `AcceptedApplicantsGradeMinimum` - Valor mínimo das classificações dos candidatos admitidos.
- `AcceptedApplicantsGradeMaximum` - Valor máximo das classificações dos candidatos admitidos.

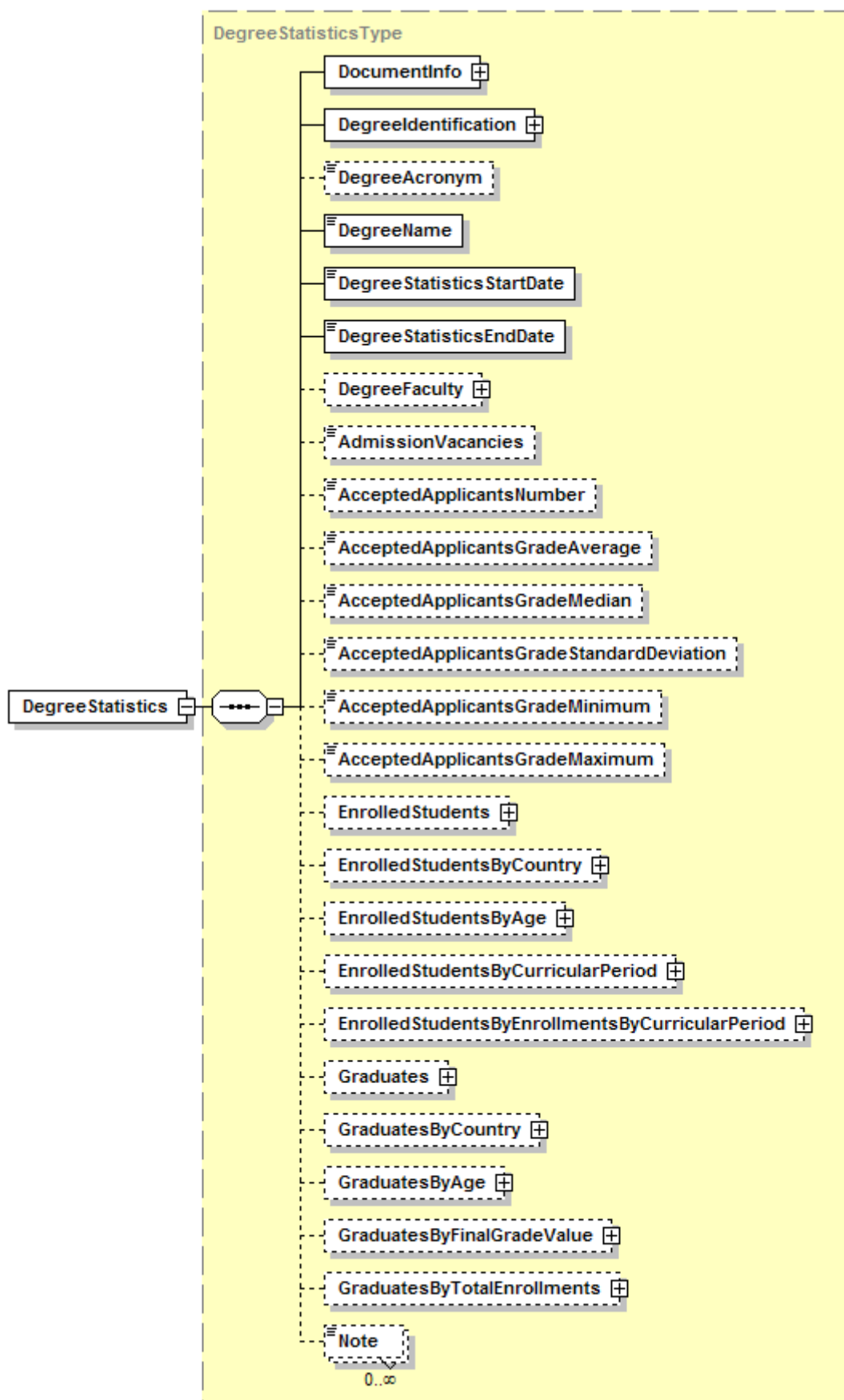


Figura 6.9: Elemento DegreeStatisticsType

- EnrolledStudents - Número de alunos inscritos.
- EnrolledStudentsByCountry - Distribuição dos alunos inscritos, por países.
- EnrolledStudentsByAge - Distribuição dos alunos inscritos, por idades.
- EnrolledStudentsByCurricularPeriod - Distribuição dos alunos inscritos, por período curricular.
- EnrolledStudentsByEnrollmentsByCurricularPeriod - Distribuição dos alunos inscritos, por número total de inscrições e para cada período curricular.
- Graduates - Número de alunos graduados.
- GraduatesByCountry - Distribuição dos alunos graduados, por países.
- GraduatesByAge - Distribuição dos alunos graduados, por idades.
- GraduatesByFinalGradeValue - Distribuição dos alunos graduados, por médias finais.
- GraduatesByTotalEnrollments - Distribuição dos alunos graduados, por número total de inscrições.

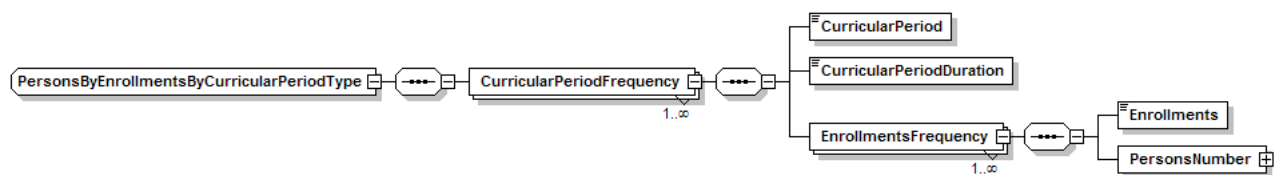


Figura 6.10: Elemento PersonsByEnrollmentsByCurricularPeriodType

Em alguns casos, elementos diferentes reutilizam o mesmo tipo. É o que acontece com EnrolledStudentsByAge e GraduatesByAge, que são ambos do tipo PersonsByAgeType.

A título de exemplo, apresenta-se com mais detalhe a estrutura dos elementos EnrolledStudentsByEnrollmentsByCurricularPeriod e GraduatesByFinalGradeValue. O primeiro é do tipo PersonsByEnrollmentsByCurricularPeriodType (ver Figura 6.10) e contém, como subelemento, um ou vários CurricularPeriodFrequency. Este elemento é composto por três subelementos obrigatórios:

- CurricularPeriod - Identificação do período curricular.
- CurricularPeriodDuration - Duração do período curricular.
- EnrollmentsFrequency - Frequência das inscrições. Representa o número de estudantes (inscritos ou graduados) por número de inscrições.

O tipo `PersonsByFinalGradeValueType` é composto por um elemento do tipo `FinalGradeValueFrequency` que pode ocorrer uma ou várias vezes (ver Figura 6.11). Este elemento contém um elemento que identifica a média final (`FinalGradeValue`) e um que reúne a distribuição das pessoas (`PersonsNumber`).

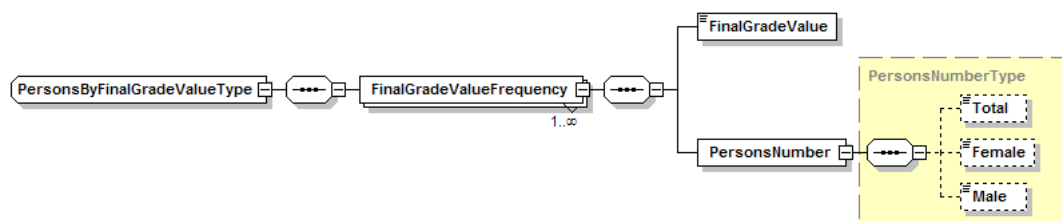


Figura 6.11: Elemento `PersonsByFinalGradeValueType`

No Apêndice C, inclui-se a listagem completa da especificação do elemento `DegreeStatisticsType`.

6.4.3 Especificação do Sistema

Diagrama de Componentes

Na vista de componentes do sistema (Figura 6.12), está representada a instituição que pretende agregar a informação (`Institution (Client)`) e a instituição que detém a informação (`Institution (Server)`). A interação parte do cliente que invoca a interface pública `getDegreeStatistics`, indicando o período lectivo que pretende.

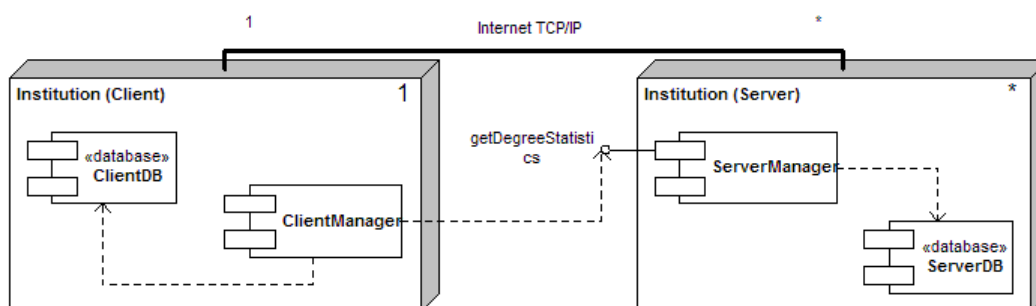


Figura 6.12: Diagrama de Componentes para a Agregação de Informação Estatística sobre Cursos

Especificação da interface:

- **getDegreeStatistics** - Recebe, via HTTP/GET, o identificador do curso. Se o pedido é válido, envia as estatísticas mais recentes do curso em formato XML.

Diagrama de Sequência

O diagrama de sequência (Figura 6.13) é muito simples uma vez que a interação tem um padrão pedido-resposta tradicional. Na figura, é ilustrada a interação com um dos servidores. Em cenários com múltiplos servidores, o cliente deve fazer a mesma interrogação a cada um dos sistemas. A ordem e periodicidade das interrogações dependem exclusivamente das opções do sistema cliente.

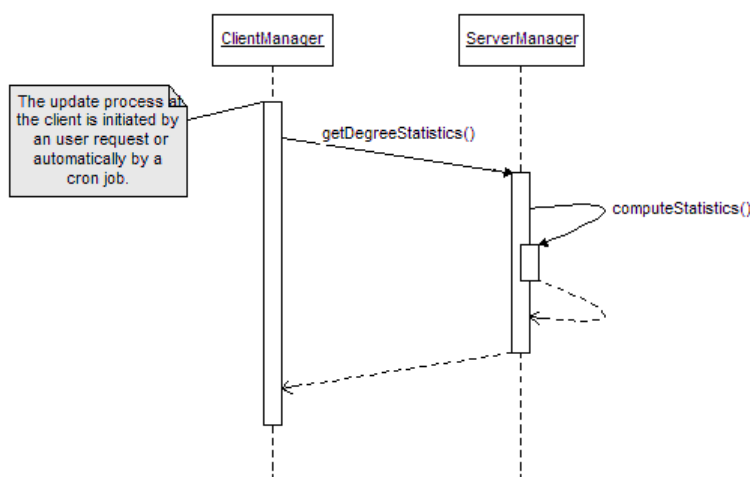


Figura 6.13: Diagrama de Sequência para a Agregação de Informação Estatística sobre Cursos

6.5 Concentração de Informação sobre Recursos Humanos

6.5.1 Identificador do Recurso Humano

Com o intuito de definir identificadores únicos para cada recurso humano e de forma semelhante ao que foi feito nos casos de estudo anteriores, propõe-se um código hierárquico com a seguinte estrutura:

Código do País - Código da Instituição - Código do Funcionário

O código do país corresponde à norma ISO 3166; o código da instituição, e no caso de Portugal, é atribuído pelo MCES; o código do funcionário é atribuído pela própria instituição. Desta forma, é possível identificar universalmente cada recurso humano.

6.5.2 Representação de Informação sobre Recursos Humanos

Para a representação de informação sobre cada recurso humano, foi definido o elemento `StaffRecord` (ver Figura 6.14). Este elemento está estruturado em cinco subelementos.

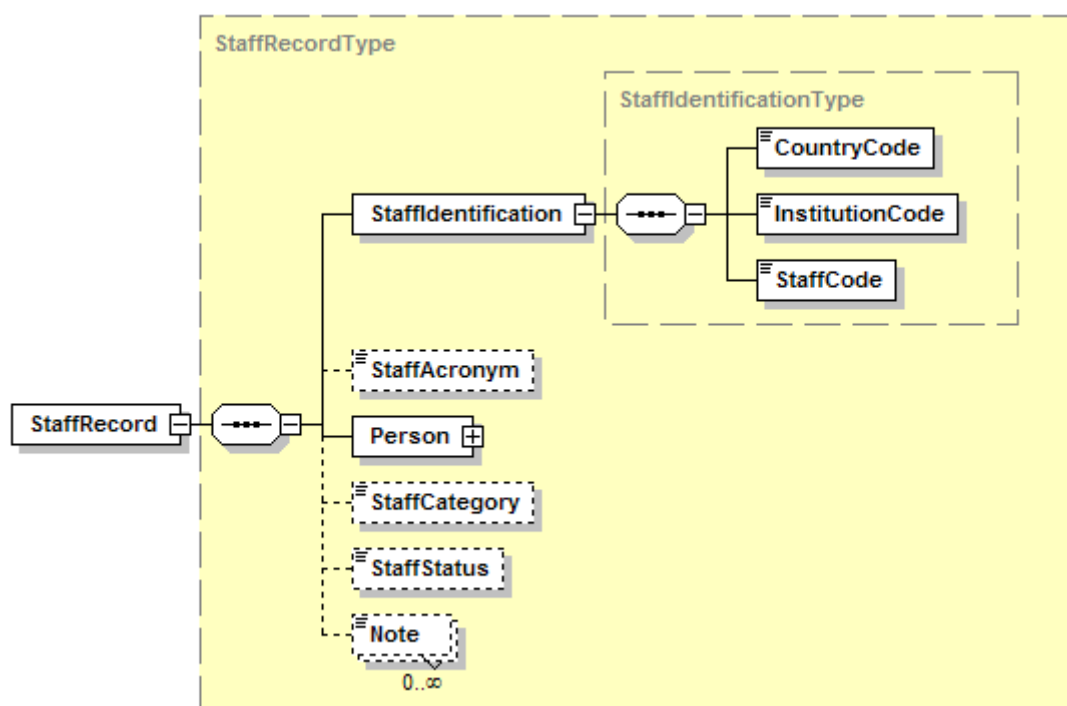


Figura 6.14: Elemento StaffRecord

- **StaffIdentification** - Reúne a identificação do país, da instituição e da pessoa na instituição.
- **StaffAcronym** - Sigla do funcionário.
- **Person** - Conjunto de informações sobre a pessoa: identificação, nome, nascimento, estado civil e contactos.
- **StaffCategory** - Categoria, no contexto da carreira, a que a pessoa pertence. Os valores possíveis dependem muito das regras definidas pela instituição e pelo governo central. Neste caso, usou-se o SiFEUP como referência na definição das opções.
- **StaffStatus** - Estado da pessoa na instituição. Também neste caso foi usado o SiFEUP como referência.
- **Note** - Observação sobre o registo. Este elemento permite armazenar informação livre que se considere relevante incluir.

No Apêndice D, encontra-se a especificação completas do elemento *StaffRecord*.

6.5.3 Especificação do Sistema

Diagrama de Componentes

O diagrama de componentes para a concentração de informação é semelhante ao da partilha de informação (Figura 6.7), com a diferença de, neste caso, existir apenas *um cliente* -

o ponto de concentração de informação. O *Institution (Server)* representa uma das instituições que detêm informação sobre recursos humanos.

O nó cliente disponibiliza a interface *updateReplica*, que permite às diversas instituições actualizar a informação concentrada nesse nó. Cada um dos servidores de informação publica três interfaces que permitem as actualizações por parte do nó central. Assim, para adicionar novos nós, basta criar as três interfaces no novo sistema e registar essa informação no ponto de concentração.

Especificação das interfaces:

- **updateReplica** - Recebe, via HTTP/POST, o identificador do funcionário (*StaffRecordId*) e o registo académico em XML (*StaffRecord*). Se o pedido é válido, actualiza a réplica local.
- **readPrimary** - Recebe, via HTTP/GET, o identificador do funcionário. Se o pedido é válido, responde com o documento XML correspondente ao registo.
- **updatePrimary** - Recebe, via HTTP/POST, o identificador do funcionário (*StaffRecordId*), a chave associada ao *lock* (*LockKey*) e o registo do funcionário em XML (*StaffRecord*). Se o pedido é válido, é actualizada a cópia primária e iniciado o processo de actualização das réplicas. Em caso de insucesso, é enviado um código de erro.
- **lockPrimary** - Recebe, via HTTP/GET, o identificador do funcionário. É efectuado um *lock* sobre o recurso. Se o procedimento é bem sucedido, é enviado na resposta um código associado ao *lock* criado. Se não é possível criar o *lock*, é enviado um código de erro.
- **unlockPrimary** - Recebe, via HTTP/GET, o identificador do funcionário e o código associado ao *lock* (*LockKey*). Se o pedido é válido, é libertado o *lock* sobre o registo.

Diagrama de Sequência

Na Figura 6.15, representa-se o diagrama de sequência para o processo de actualização de informação a partir do nó central. O processo é iniciado quando existe um pedido de actualização. Como é necessário garantir que a informação disponível é a mais recente, é feito um *bloqueio* ao recurso que se vai alterar e uma actualização dos dados no cliente. Depois de concluída a actualização, é feita a propagação para o servidor e os bloqueios são libertados. No final deste processo, e de forma assíncrona, o servidor inicia a actualização da réplica existente no cliente.

Como se indica na figura, existe um período durante o qual o nó central não vai ter na base de dados a cópia mais recente do recurso alterado. Uma nova invocação da interface *readPrimary*, imediatamente depois da actualização, evitaria esta situação.

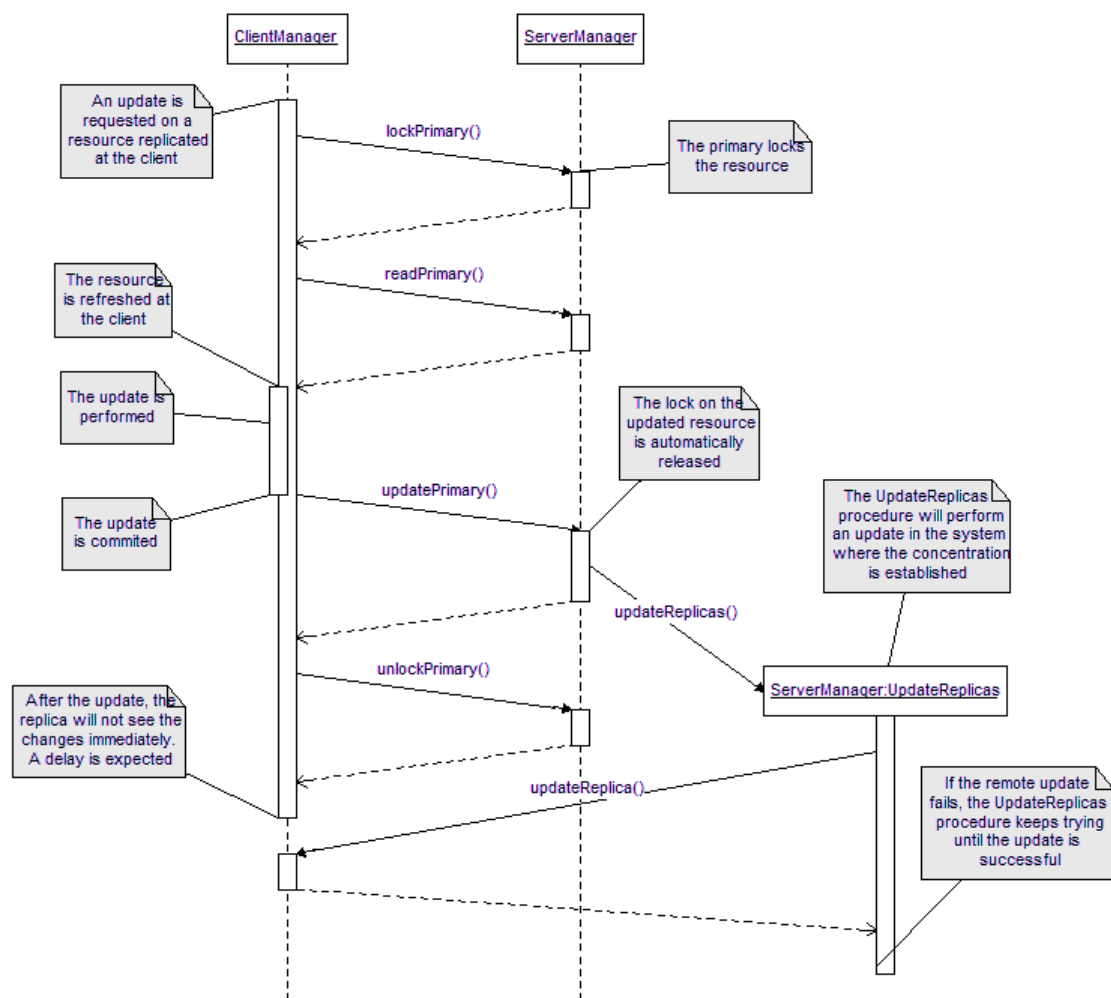


Figura 6.15: Diagrama de Sequência para a Concentração de Informação sobre Recursos Humanos

O processo de actualização de informação a partir de um dos servidores terá um diagrama de sequência semelhante ao que foi apresentado na Figura 6.8. Com a diferença de, neste caso, existir apenas uma réplica para actualizar.

6.6 Difusão de Notícias da Universidade

6.6.1 Identificador da Notícia

Para a implementação de identificadores únicos a atribuir a cada notícia, e seguindo a mesma estratégia adoptada nos outros casos de estudo, propõe-se um código hierárquico com a seguinte estrutura:

Código do País - Código da Instituição - Código do Notícia

O código do país corresponde à norma ISO 3166; o código da instituição, e no caso de Portugal, é atribuído pelo MCES; o código da notícia é atribuído pela própria instituição. Desta forma, é possível identificar universalmente cada notícia.

6.6.2 Representação de Notícias

O elemento `NewsItem`, ilustrado na Figura 6.16, representa um item noticioso. É composto por cinco elementos:

- `NewsItemIdentification`: Identificação do item noticioso. Reúne informação sobre o país, a instituição e o item.
- `Title`: O título da notícia.
- `Date`: A data de publicação da notícia.
- `ExpirationDate`: A data de expiração da notícia.
- `Description`: A descrição da notícia.
- `Author`: O autor ou o responsável pela notícia.
- `Category`: A categoria a que pertence a notícia. Este elemento pode existir várias vezes, permitindo associar uma notícia a múltiplas categorias.
- `URL`: Endereço Web permanente da notícia.

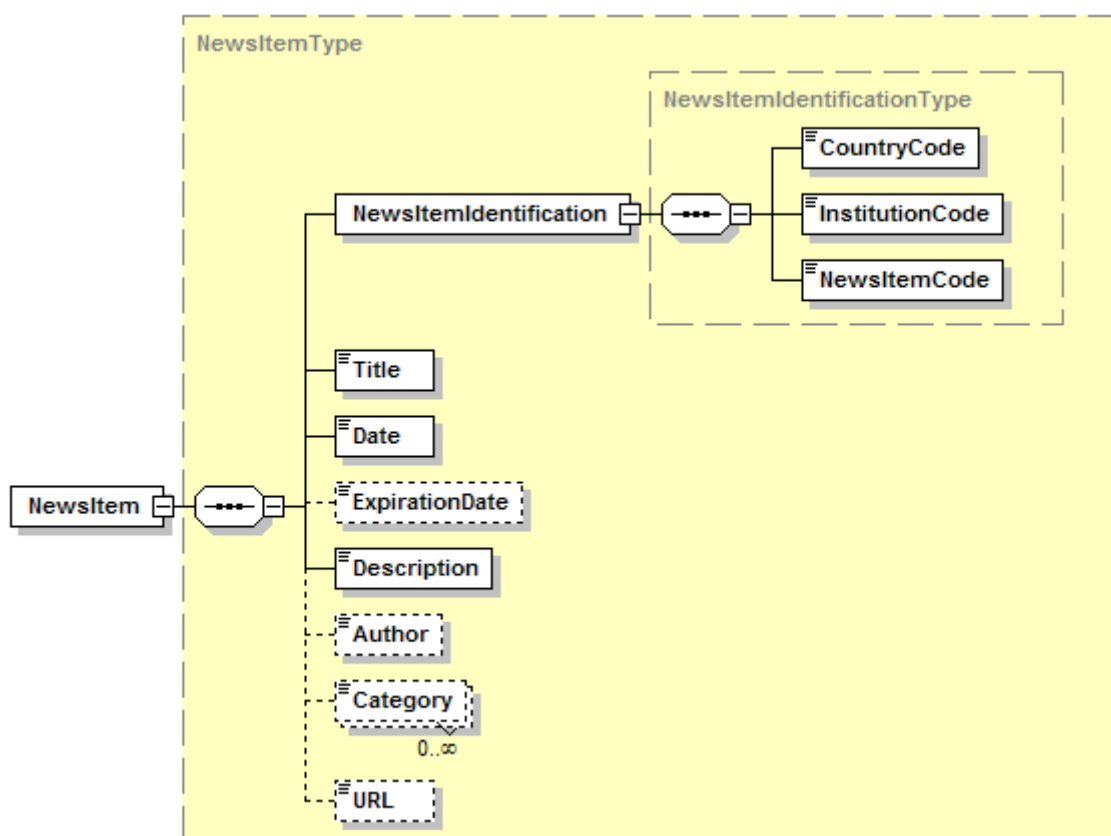


Figura 6.16: Elemento NewsItem

A especificação completa do elemento NewsItem está disponível no apêndice E.

No caso particular da difusão pública e para além da disponibilização dos itens noticiosos no formato aqui especificado, sugere-se a adopção de normas públicas que permitam uma melhor interoperabilidade com os sistemas existentes. A justificação advém do facto de, no caso da difusão pública, se procurar potenciar a integração com sistemas externos.

Assim, e considerando o panorama actual da difusão de informação noticiosa na WWW, opta-se pelo formato RSS 2.0 [101]. O RSS (*Really Simple Syndication*) é um dialecto XML desenvolvido inicialmente pela Netscape para a representação de alterações a sítios Web. Actualmente, é usado para a publicação de notícias por parte de empresas como a BBC [3], Reuters [81] e o *The Washington Post* [79]. Um outro argumento a favor da adopção deste formato é a existência de bibliotecas específicas para RSS nas principais linguagens de programação.

6.6.3 Especificação do Sistema

Diagrama de Componentes

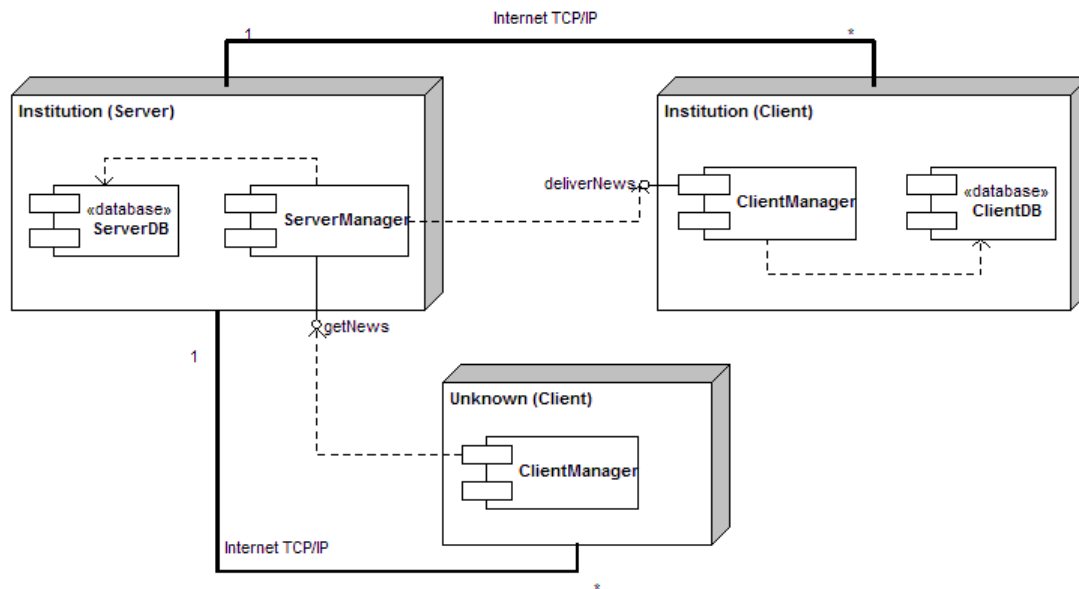


Figura 6.17: Diagrama de Componentes para a Difusão de Notícias da Universidade

Na Figura 6.17 é apresentado o diagrama de componentes do sistema. A Reitoria da Universidade é representada pelo nó *Institution (Server)* e os nós *Institution (Client)* e *Unknown (Client)* representam, respectivamente, uma instituição da universidade (difusão controlada) e uma entidade desconhecida (difusão pública).

O servidor publica a interface *getNews*, que é usada por parte dos clientes públicos para aceder à lista das últimas notícias. Para a difusão controlada, o servidor utiliza a interface *deliverNews*, disponibilizada por cada uma das instituições envolvidas na comunidade.

Especificação das interfaces:

- **deliverNews** - Recebe, via HTTP/POST, a notícia em formato XML (*NewsItem*). Se o pedido é válido, a notícia é armazenada localmente.
- **getNews** - Recebe pedidos via HTTP/GET e responde com a lista das últimas notícias em formato RSS.

Diagramas de Sequência

Nas Figuras 6.18 e 6.19, estão representados os diagramas de sequência para os dois casos de estudo seleccionados. No primeiro caso, a execução é despoletada sempre que existe um novo item para difundir. O servidor selecciona os clientes com base em critérios pré-definidos e faz o envio do item para cada um.

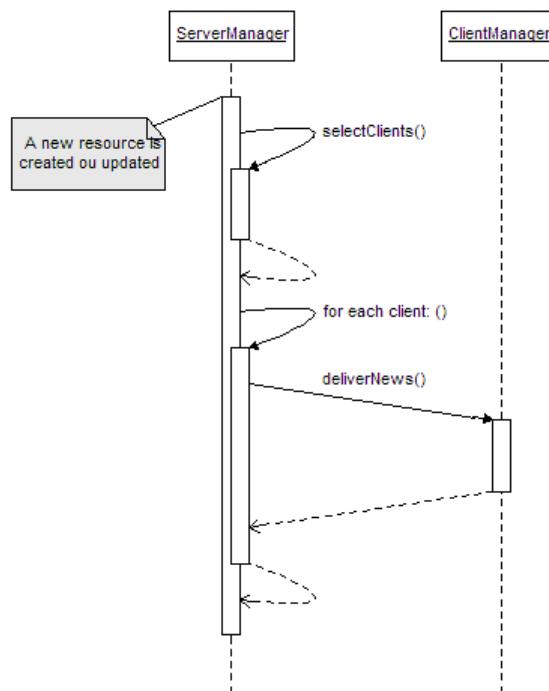


Figura 6.18: Diagrama de Sequência para a Difusão Controlada de Notícias

No cenário de difusão pública, a interação é iniciada pelo cliente. O servidor responde aos pedidos, enviando a lista das últimas notícias.

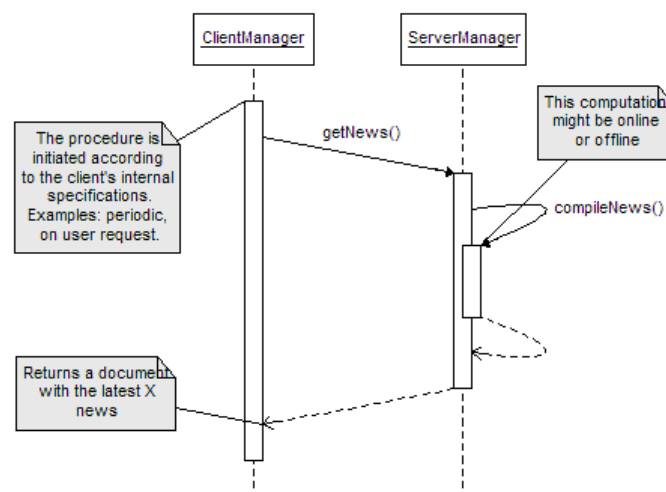


Figura 6.19: Diagrama de Sequência para a Difusão Pública de Notícias

6.7 Protótipos

Tendo como objectivo analisar a viabilidade prática das soluções propostas, foram desenvolvidos vários protótipos. Estes protótipos estão divididos em dois grupos, de acordo com o que se pretendeu avaliar: representação da informação e mecanismos de interacção.

Em relação ao primeiro caso, procurou-se testar a adequação das especificações propostas para representação de informação em situações reais. Para tal, recorreu-se à informação existente no SiFEUP e foi desenvolvido um protótipo que permitiu extrair representações do registo académico de cada aluno no formato proposto. O sistema de informação da FEUP é construído sobre o SGBD Oracle. Utilizando as funções disponíveis para produção de XML a partir de interrogações SQL à base de dados, foi possível extrair um documento XML com os dados necessários. Aplicando a esta saída uma transformação XSL produziu-se o documento final. Este cenário é ilustrado na Figura 6.20.

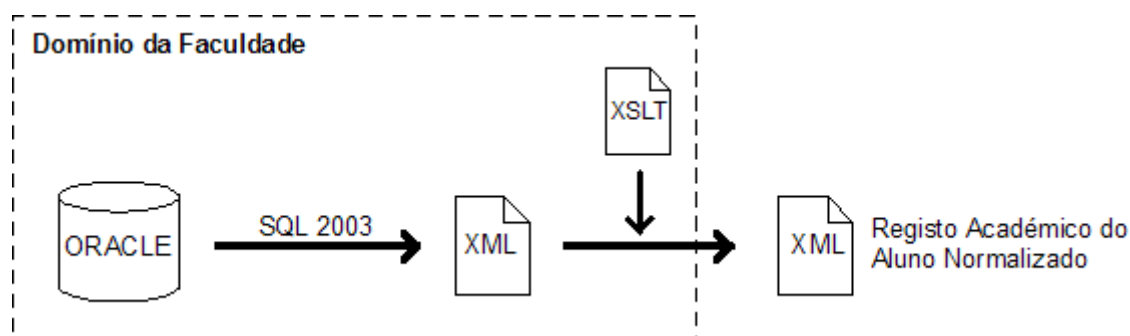


Figura 6.20: Arquitectura do protótipo relativo à representação de informação

Em relação aos mecanismos de interacção, foram implementados protótipos para cada um dos padrões. Os protótipos foram desenvolvidos em PHP [51] e os testes realizados sobre três servidores geograficamente dispersos. Estes testes permitiram validar as arquitecturas propostas e, em particular, as tecnologias adoptadas (XML, URI e HTTP). Também foi possível testar com sucesso a implementação de mecanismos de controlo de acessos, recorrendo ao esquema de autenticação básica do HTTP [80].

Teria sido interessante explorar em maior detalhe os protótipos desenvolvidos, em particular com a realização de testes de carga e integração parcial com os sistemas em produção, no entanto o factor tempo e condicionantes deste tipo de trabalho não o permitiram.

Capítulo 7

Conclusões

Neste capítulo, é apresentado um sumário das conclusões obtidas e são sugeridas algumas perspectivas de trabalho futuro.

7.1 Conclusões Gerais

Neste trabalho, desenvolveu-se uma investigação sobre as principais alternativas para a interoperabilidade entre sistemas de informação universitários. Foram identificados e categorizados os padrões de interacção mais comuns e foram propostas soluções para os principais cenários encontrados. A construção de protótipos permitiu validar, numa primeira fase, as abordagens propostas.

As principais contribuições desta dissertação para a área em estudo podem ser sintetizadas em quatro itens:

- *Síntese do estado da arte em relação às principais tecnologias de apoio à interoperabilidade.* É feita uma revisão sistemática e transversal das principais questões que, de um ponto de vista tecnológico, condicionam a tomada de decisão na construção de sistemas interoperáveis.
- *Estruturação e caracterização dos padrões de interoperabilidade em ambientes universitários.* Para a definição do problema, foram identificados dois eixos (vertical e horizontal) e caracterizados quatro padrões de interacção: partilha, agregação, concentração e difusão.
- *Proposta de especificações XML para a representação de informação no contexto académico.* Concretamente, foram elaboradas propostas para a representação do registo académico do aluno, de estatísticas sobre os cursos, de informação sobre os recursos humanos e de notícias produzidas pelas instituições. Neste contexto, foi também proposta uma estratégia para a definição de identificadores únicos associados a esta informação.
- *Proposta de especificações UML para a implementação de mecanismos de interacção.* Com base nos padrões de interacção identificados, foram propostas especificações UML para a partilha, agregação, concentração e difusão de informação.

A interoperabilidade entre sistemas de informação é uma área complexa em que se cruzam muitas disciplinas. Mesmo considerando apenas o aspecto tecnológico, é evidente o recurso a áreas tão distintas como as redes de comunicação de dados, a representação de informação e a programação de sistemas distribuídos. Por outro lado, a construção de sistemas distribuídos é intrinsecamente diferente de sistemas monolíticos; questões como a latência, a concorrência ou as falhas parciais aumentam a complexidade destes sistemas.

Nos últimos tempos, esta área tem sido alvo de crescente atenção; a produção de especificações e a constituição de consórcios empresariais tem crescido fortemente. Se por um lado, estas iniciativas resultam em desenvolvimentos tecnológicos importantes; por outro, aumentam a complexidade da decisão devido à proliferação de soluções para o mesmo fim.

Muitas das tecnologias actualmente propostas não atingiram ainda a maturidade necessária para implementações reais. O SOAP é um exemplo de uma norma onde implementações produzidas por diferentes fabricantes são incompatíveis entre si. Neste trabalho, optou-se por soluções baseadas em tecnologias bem estabelecidas e largamente difundidas como o HTTP, os URL e o XML.

Um outro aspecto que importa referir, é o facto de este trabalho ter permitido constatar que o XML é uma tecnologia poderosa e largamente difundida. O leque de ferramentas existentes e a sua generalizada adopção por parte da indústria confirmam o estatuto de *standard de facto*.

Uma das estratégias seguidas neste trabalho foi o desenvolvimento de soluções simples com pouco impacto sobre os sistemas existentes. Por exemplo, as interfaces propostas para os mecanismos de interacção, baseiam-se nos verbos HTTP. Esta opção tem como fundamento o facto de ser mais interessante *estabelecer acordos simples que englobem a maioria dos participantes, do que acordos exaustivos e muito detalhados envolvendo apenas um leque reduzido de participantes*.

Concluindo, a interoperabilidade atinge-se aderindo a normas comuns. Neste caso, optou-se por uma solução que usa tecnologias largamente difundidas: tecnologias Web e XML. O trabalho realizado permitiu validar estas tecnologias na implementação das soluções propostas. *A chave para a interoperabilidade é a normalização*. Um factor imprescindível ao sucesso de qualquer iniciativa nesta área será o apoio por parte dos centros de poder.

7.2 Perspectivas de Trabalho Futuro

Sistematizam-se aqui alguns tópicos que podem ser explorados em trabalhos futuros:

- Levar a cabo esforços que permitam conduzir à adopção institucional das especificações aqui propostas. O empenho dos órgãos de gestão é um factor fundamental no sucesso de políticas que envolvem diferentes grupos (unidades, instituições ou universidades). Assim, para além da interoperabilidade técnica aqui abordada, é importante apostar na *interoperabilidade política/humana*, relacionada com a dimensão organizacional, as relações de poder e as culturas.
- Explorar as possibilidades de adopção de ontologias, existentes ou a desenvolver, na área académica. As ontologias permitem formalizar conceitos e as respectivas relações em domínios específicos.
- No Capítulo 3 foi apresentado o conceito de Níveis de Interoperabilidade. O trabalho aqui apresentado enquadra-se no Grau 3 da arquitectura NC3TA - “troca automática de dados sobre sistemas com modelos de partilha comuns”. Sugere-se a prossecução do trabalho nesta área, com a exploração de soluções que permitam um maior grau de interoperabilidade. Buscando atingir níveis que permitam, por exemplo, a manipulação interactiva dos dados, a partilha de aplicações ou a interpretação “universal” da informação.
- No caso particular da difusão de informação, a possibilidade de gerir dinamicamente a subscrição e o controlo de acessos das instituições interessadas em receber informação constituiria um desenvolvimento interessante. Na proposta aqui apresentada, esta informação é introduzida manualmente, cada instituição define *a priori* quais os receptores e quais os tópicos das mensagens.

- A segurança é um dos aspectos mais importantes na construção de procedimentos digitais. Por oposição ao desenvolvimento de soluções específicas para cada caso, uma solução vulgar é a definição de plataformas partilhadas de forma a potenciar a interoperabilidade. Assim, sugere-se o desenvolvimento de uma infraestrutura de chaves públicas (PKI) ao nível da universidade, de forma a garantir serviços básicos de segurança na transmissão de dados: autenticação, integridade e confidencialidade.

Bibliografia

- [1] Harald T. Alvestrand. IETF Policy on Character Sets and Languages. RFC 2277, Internet Engineering Task Force, 1998. Available from: <http://www.ietf.org/rfc/rfc2277.txt> [cited 27-05-2004].
- [2] Paul Bacsich, Andy Heath, Paul Lefrere, Paul Miller, and Kevin Riley. The Standards Fora for Online Education. *D-Lib Magazine*, 5(12), 1999. Available from: <http://www.dlib.org/dlib/december99/12miller.html>.
- [3] BBC. RSS Feed (Really Simple Syndication) [online]. 2004 [cited 03-05-2004]. Available from: <http://news.bbc.co.uk/1/hi/help/rss/default.stm>.
- [4] Tim Berners-Lee. Information Management: A Proposal. Technical report, CERN, mar 1989. Technical Report. Available from: <http://www.w3.org/History/1989/proposal.html> [cited 28-11-2003].
- [5] Tim Berners-Lee, Roy T. Fielding, and Larry Masinter. Uniform Resource Identifiers (URI): Generic Syntax. RFC 2396, Internet Engineering Task Force, 1998. Available from: <http://www.ietf.org/rfc/rfc2396.txt> [cited 26-05-2004].
- [6] Paul Beynon-Davies. *Information Systems - An Introduction to Informatics in Organizations*. Palgrave, Bath, 2002.
- [7] Chris Britton. *IT Architectures and Middleware*. Addison-Wesley, 2000.
- [8] Mike Burner. The Deliberate Revolution - Creating Connectedness with XML Web Services. *ACM Queue*, pages 29–37, 2003.
- [9] Christopher Bussler. B2B Protocol Standards and their Role in Semantic B2B Integration Engines. *Bulletin of the Technical Committee on Data Engineering*, 24(1):3–11, mar 2001.
- [10] CEN/ISSS. Interoperability frameworks for exchange of information between diverse management systems. *Learning Technology eBrochure*, (5), jan 2003. Available from: <http://www.cenorm.be/cenorm/businessdomains/businessdomains/informationssocietystandardizationsystem/elearning/learning+technologies+workshop/ebrochurenumber5.pdf>.

- [11] Alexandre Coimbra and António Rito Silva. Support Process Patterns in Higher Education. In *EUNIS 2003*. EUNIS, 2003.
- [12] ANSI ASC X12C Communications and Controls Subcommittee. ASC X12 Reference Model for XML Design. Technical report, DISA Inc., 2002. Report Number: ASC X12C/2002-61. Available from: http://www.x12.org/x12org/comments/X12Reference_Model_For_XML_Design.pdf [cited 10-12-2003].
- [13] World Wide Web Consortium. W3C XML Encryption Working Group [online]. 2003 [cited 10-04-2003]. Available from: <http://www.w3.org/Encryption/2001/>.
- [14] World Wide Web Consortium and Internet Engineering Task Force. IETF/W3C XML-DSig Working Group [online]. 2003 [cited 10-09-2003]. Available from: <http://www.w3.org/Signature/>.
- [15] Roger L. Costello. XML Schemas: Best Practices [online]. 2003 [cited 10-07-2003]. Available from: <http://xfront.com/BestPracticesHomepage.html>.
- [16] George Coulouris, Jean Dollimore, and Tim Kindberg. *Distributed Systems: Concepts and Design, 3rd Edition*. Addison-Wesley, 2001.
- [17] Postsecondary Electronic Standards Council. Work Products [online]. 2002 [cited 02-05-2003]. Available from: <http://www.pescxml.org/xmlprod.asp?page=workprod&sub=workproducts>.
- [18] Postsecondary Electronic Standards Council. PESC - home page [online]. 2003 [cited 15-03-2003]. Available from: <http://www.standardscouncil.org>.
- [19] Postsecondary Electronic Standards Council. PESC - XML Postsecondary Transcript Schema [online]. 2003 [cited 10-10-2003]. Available from: <http://www.standardscouncil.org/XMLPostTranscript.asp>.
- [20] Postsecondary Electronic Standards Council. XML forum FAQ [online]. 2003 [cited 09-04-2003]. Available from: <http://www.pescxml.org/xmlfaq.asp?page=faq>.
- [21] Postsecondary Electronic Standards Council. XML Forum for Education [online]. 2003 [cited 15-03-2003]. Available from: <http://www.pescxml.org>.
- [22] Postsecondary Electronic Standards Council. XML Technical Specification for Higher Education. Technical report, Postsecondary Electronic Standards Council, may 2003. Available from: <http://standardscouncil.org/docs/XML%20Forum%20Tech%20Specification%20v%202.5.doc>.
- [23] Postsecondary Electronic Standards Council. XML Technical Specification for Higher Education. Specification, Postsecondary Electronic Standards Council, 2003. Available from: <http://www.pescxml.org/docs/XMLForumTechSpecv01.pdf>.

- [24] Ministério da Ciência e do Ensino Superior. Acesso ao Ensino Superior 2003 - Índices de Cursos (por estabelecimento e curso) [online]. 2003 [cited 20-11-2003]. Available from: <http://www.acessoensinosuperior.pt/indest.asp>.
- [25] Ministério da Ciência e do Ensino Superior. Observatório da Ciência e do Ensino Superior [online]. 2003 [cited 01-12-2003]. Available from: <http://www.oces.mces.pt>.
- [26] Organização das Nações Unidas. Electronic Business eXtensible Mark-up Language (ebXML) [online]. 2003 [cited 06-06-2003]. Available from: <http://www.unece.org/cefact/ebxml/specifications.htm>.
- [27] DCMI. DCMI Education Working Group [online]. 2003 [cited 15-03-2003]. Available from: <http://dublincore.org/groups/education/>.
- [28] Pavan Deolasee, Amol Katkar, Ankur Panchbudhe, Krithi Ramamritham, and Prashant Shenoy. Adaptive push-pull: disseminating dynamic web data. In *Proceedings of the tenth international conference on World Wide Web*, pages 265–274. ACM Press, 2001.
- [29] Universidade do Porto. Universidade do Porto - home page [online]. 2003 [cited 27-05-2003]. Available from: <http://www.up.pt>.
- [30] EDUCAUSE. What is EDUCAUSE? [online]. 2003 [cited 2003-02-20]. Available from: <http://www.educause.edu/defined.html>.
- [31] EDUCAUSE and Internet2 eduPerson Task Force. eduPerson Object Class [online]. 2003 [cited 15-03-2003]. Available from: <http://www.educause.edu/eduperson/>.
- [32] David L. Eisler. Campus Portals: Supportive Mechanisms for University Communication, Collaboration, and Organizational Change. *Journal of Computing in Higher Education*, 13(1):3–24, 2001. Available from: <http://faculty.weber.edu/deisler/journal.htm>.
- [33] Wolfgang Emmerich. Software Engineering and Middleware: a Roadmap. In *Proceedings of the conference on The future of Software Engineering*, pages 117–129. ACM Press, 2000.
- [34] Patrick Th. Eugster, Pascal A. Felber, Rachid Guerraoui, and Anne-Marie Kermarrec. The Many Faces of Publish/Subscribe. *ACM Computing Surveys*, 35(2):114–131, jun 2003.
- [35] EUNIS. EUNIS Website [online]. 2003 [cited 2003-02-20]. Available from: <http://www.eunis.org>.
- [36] União Europeia. Declaração de Bolonha, 1999. Available from: http://www.bologna-berlin2003.de/pdf/bologna_declaration.pdf.

- [37] União Europeia. Sistema Europeu de Transferência e Acumulação de Créditos de Curso (ECTS) [online]. 2003 [cited 05-01-2004]. Available from: http://europa.eu.int/comm/education/programmes/socrates/ects_pt.html.
- [38] David C. Fallside. XML Schema Part 0: Primer. Recommendation, World Wide Web Consortium, 2001. Available from: <http://www.w3.org/TR/xmlschema-0/> [cited 16-05-2003].
- [39] Roy T. Fielding, James Gettys, Jeffrey C. Mogul, Henry F. Nielsen, Larry Masinter, Paul J. Leach, and Tim Berners-Lee. Hypertext Transfer Protocol - HTTP/1.1. RCF 2616, Internet Engineering Task Force, 1999. Available from: <http://www.ietf.org/rfc/rfc2616.txt> [cited 26-05-2004].
- [40] Roy T. Fielding and Richard N. Taylor. Principled Design of the Modern Web Architecture. *ACM Transactions on Internet Technology*, 2(2):115–150, 2002.
- [41] International Organization for Standards. Maintenance agency for iso 3166 country codes [online]. 2003 [cited 18-11-2003]. Available from: <http://www.iso.ch/iso/en/prods-services/iso3166ma/index.html>.
- [42] Michael Franklin and Stanley Zdonik. A framework for scalable dissemination-based systems. In *Proceedings of the 12th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 94–105. ACM Press, 1997.
- [43] Michael J. Franklin, Michael J. Carey, and Miron Livny. Transactional Client-Server Cache Consistency: Alternatives and Performance. *ACM Transactions on Database Systems*, 22(3):315–363, 1997.
- [44] Michael J. Franklin, Björn Thór Jónsson, and Donald Kossmann. Performance Tradeoffs for Client-Server Query Processing. In *Proceedings of the 1996 ACM SIGMOD international conference on Management of data*, pages 149–160. ACM Press, 1996.
- [45] John Franks, Phillip M. Hallam-Baker, Jeffery L. Hostetler, Scott D. Lawrence, Paul J. Leach, Ari Luotonen, and Lawrence C. Stewart. HTTP Authentication: Basic and Digest Access Authentication. RFC 2617, Internet Engineering Task Force, 1999. Available from: <http://www.ietf.org/rfc/rfc2617.txt> [cited 26-05-2004].
- [46] Ursula Fuller, Cornelia Helbling, and Roger Cooley. The Potential Of Web-Based Suggestion Schemes In A University Setting. In *EUNIS 2002*. EUNIS, FEUP Edições, 2002.
- [47] Kurt Geihs. Middleware Challenges Ahead. *IEEE Computer*, 34(6):24–31, jun 2001.

- [48] Jim Gray, Pat Helland, Patrick O'Neil, and Dennis Shasha. The dangers of replication and a solution. In *Proceedings of the 1996 ACM SIGMOD international conference on Management of data*, pages 173–182. ACM Press, 1996.
- [49] C4ISR Architecture Working Group. *Levels of Information Systems Interoperability*, 1998.
- [50] ISSC NATO Open Systems Working Group. *NATO C3 Technical Architecture - Volume 2: Architectural Descriptions and Models*, 2003.
- [51] The PHP Group. PHP: Hypertext Preprocessor [online]. 2004 [cited 27-05-2004]. Available from: <http://www.php.net>.
- [52] Hugo Haas. Designing the architecture for Web Services. In *WWW2003. W3C*, 2003. Available from: <http://www.w3.org/2003/Talks/0521-hh-wsa/>.
- [53] Franz Haselbacher. Design and operation of a WEB-databased university-information-management-system. In *EUNIS 2002*. EUNIS, FEUP Edições, 2002.
- [54] HL7. Health Level 7 [online]. 2003 [cited 05-03-2003]. Available from: <http://www.hl7.org>.
- [55] HUMANE. Seminar - International Students: Exchange or Trade? [online]. 2003 [cited 25-11-2003]. Available from: http://www.humane.eu.org/5_down_lund_2.doc.
- [56] IMS. IMS Global Learning Consortium, Inc. [online]. 2003 [cited 15-03-2003]. Available from: <http://www.imsglobal.org>.
- [57] Internet2/MACE. Directory of Directories for Higher Education [online]. 2003 [cited 15-03-2003]. Available from: <http://middleware.internet2.edu/dodhe/>.
- [58] Internet2/MACE. Shibboleth Project [online]. 2003 [cited 15-03-2003]. Available from: <http://shibboleth.internet2.edu>.
- [59] Internet2/MI. Internet2 Middleware [online]. 2003 [cited 04-04-2003]. Available from: <http://middleware.internet2.edu>.
- [60] Carl Jacobson. Institutional Information Portals. *EDUCAUSE Review*, pages 58–59, 2000. Available from: <http://www.educause.edu/pub/er/erm00/articles004/horizons.pdf>.
- [61] Donald Kossmann. The State of the Art in Distributed Query Processing. *ACM Computing Surveys*, 32(4):422–469, dec 2000.
- [62] Donald Kossmann, Michael J. Franklin, Gerhard Drasch, and Wig Ag. Cache investment: integrating query optimization and distributed data placement. *ACM Transactions on Database Systems*, 25(4):517–558, 2000.

- [63] Kenneth C. Laudon and Jane P. Laudon. *Management Information Systems*. Prentice Hall, New Jersey, 2002.
- [64] Bin Ling, Colin Allison, and Malcolm Bain. Analyse, Model, Improve, Deriving Value Added Services through Process Modeling. In *EUNIS 2002*. EUNIS, FEUP Edições, 2002.
- [65] Ling Liu, Ling Ling Yan, and M. Tamer Ozsu. Interoperability in Large-scale Distributed Information Delivery Systems. In *NATO Advanced Study Institute on Workflow Management Systems and Interoperability*. NATO, 1997.
- [66] Peter Mederly. Annual Eunis Report on University Information Systems in Europe. In *EUNIS97 Conference*. EUNIS, 1997. Available from: <http://www.lmcp.jussieu.fr/eunis/html3/congres/EUNIS97/papers/052201.html>.
- [67] Brahim Medjahed, Boualem Benatallah, Athman Bouguettaya, Anne H. Ngu, and Ahmed K. Elmagarmid. Business-to-business interactions: issues and enabling technologies. *The VLDB Journal*, 12(1):59–85, 2003.
- [68] Paul Miller. Interoperability. What is it and Why should i want it? *Ariadne*, (24), 2000. Available from: <http://www.ariadne.ac.uk/issue24/interoperability/>.
- [69] Janina Mincer-Daszkiewicz. Student Management Information System for Polish Universities. In *EUNIS 2002*. EUNIS, FEUP Edições, 2002.
- [70] Henry Mintzberg. *Estrutura e Dinâmica das Organizações*. Publicações Dom Quixote, Lisboa, 1995.
- [71] NCES. SPEEDE/ExPRESS Home Page [online]. 2003 [cited 15-03-2003]. Available from: <http://nces.ed.gov/edi/speedeExp.asp>.
- [72] Anoop Ninan, Purushottam Kulkarni, Prashant Shenoy, Krithi Ramamritham, and Renu Tewari. Cooperative Leases: Scalable Consistency Maintenance in Content Distribution Networks. In *WWW2002*. ACM Press, 2002.
- [73] OASIS. Open Architecture and Schools in Society [online]. 2003 [cited 15-03-2003]. Available from: <http://oasis.cnice.mecd.es>.
- [74] Florence Olsen. Sharing the Code [online]. 2003 [cited 10-08-2003]. Available from: <http://chronicle.com/free/v49/i47/47a03101.htm>.
- [75] OMG. Introduction To OMG Specifications: CORBA [online]. 2002 [cited 04-01-2003]. Available from: <http://www.omg.org/gettingstarted/specintro.htm#CORBA>.
- [76] Esther Pacitti and Eric Simon. Update propagation strategies to improve freshness in lazy master replicated databases. *The VLDB Journal*, 8(3-4):305–318, 2000.

- [77] C. M. Parker and P. M. C. Swatman. Encouraging SME Acceptance of EDI: An Educational Approach. In *8th International EDI-IOS Conference*, pages 27–46. EDI-IOS, jun 1995. Available from: <http://citeseer.nj.nec.com/36012.html>.
- [78] Kestutis Pocius, Aleksandras Targamazde, Linas Vingelis, and Albertas Zalys. Designing Corporate Information System for cooperation and management of Lithuanian HE&R institutions. In *EUNIS 2002*. EUNIS, FEUP Edições, 2002.
- [79] The Washington Post. RSS Feeds [online]. 2004 [cited 03-05-2004]. Available from: <http://www.washingtonpost.com/wp-adv/rss/>.
- [80] Eric Rescorla. HTTP Over TLS. RFC 2818, Internet Engineering Task Force, 2000. Available from: <http://www.ietf.org/rfc/rfc2818.txt> [cited 26-05-2004].
- [81] Reuters. Reuters RSS [online]. 2004 [cited 09-05-2004]. Available from: <http://www.reuters.com/newsrss.jhtml>.
- [82] Lígia Ribeiro and José Marques dos Santos, editors. *The Changing Universities: The Challenge of New Technologies*. EUNIS, FEUP Edições, 2002.
- [83] David Ritter. The Middleware Muddle. *SIGMOD Record*, 27(4):86–93, 1998.
- [84] Manuel Rodríguez-Martínez and Nick Roussopoulos. MOCHA: A Self-Extensible Database Middleware System for Distributed Data Sources. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 213–224. ACM Press, 2000.
- [85] Alenka Rozanec, Rok Rupnik, Marki Bajec, and Marjan Krisper. IS Strategic plan for the University of Ljubljana: Experiences and Results. In *EUNIS 2002*. EUNIS, FEUP Edições, 2002.
- [86] SCANet. Norma 1.01/02E - Identificación Normalizada del Código Unificado de Estudiante. Technical report, SCANet, mar 2002. Available from: <http://scanet.udl.es/spa/normas/Norma-1.1-02E-VG.pdf>.
- [87] SCANet. Norma 1.01/02E - Identificación Normalizada del Código Unificado de Estudiante. Technical report, SCANet, mar 2002. Available from: <http://scanet.udl.es/spa/normas/Norma-2.1-02E-VG.pdf>.
- [88] SCANet. Norma 4.01/03E - Homologación de Datos de Expedientes de Primer y Segundo Ciclo para Intercambios, traslados y certificaciones académicas. Technical report, SCANet, jun 2003. Available from: <http://scanet.udl.es/spa/normas/Norma-4.1-03E.pdf>.
- [89] SCANet. SCANet [online]. 2003 [cited 30-05-2003]. Available from: <http://www.scanet.udl.es>.

- [90] SCANet. SCANet - Fases [online]. 2003 [cited 24-11-2003]. Available from: <http://scanet.udl.es/spa/fases/index.html>.
- [91] SCANet. SCANet - Nuestra misión [online]. 2003 [cited 23-11-2003]. Available from: <http://scanet.udl.es/spa/definicion/index.html>.
- [92] Tony Shaw, Anne Strachan, Gerry McCauley, and Lynn McCrae. The portal as the framework for the information strategy? In *EUNIS 2002*. EUNIS, FEUP Edições, 2002.
- [93] SIF. Schools Interoperability Framework [online]. 2003 [cited 15-03-2003]. Available from: <http://www.sifinfo.org>.
- [94] Doug Stacey. Replication: DB2, Oracle, or Sybase? *SIGMOD Record*, 21(1):95–101, 1995.
- [95] IEEE Computer Society Standards. IEEE Learning Technology Standards Committee [online]. 2003 [cited 15-03-2003]. Available from: <http://ltsc.ieee.org>.
- [96] Stefan Tai and Isabelle Rouvellou. Strategies for Integrating Messaging and Distributed Object Transactions. In *IFIP/ACM International Conference on Distributed systems platforms*, pages 308–330. Springer-Verlag New York, Inc., 2000.
- [97] Andrew S. Tanenbaum and Maarten Van Steen. *Distributed Systems: Principles and Paradigms*. Prentice-Hall, New Jersey, 2002.
- [98] Leonor Teixeira, Carlos Ferreira, and Rui Santiago. Managing Academic Services Through the Web: an information support system for graduated programs. In *EUNIS 2002*. EUNIS, FEUP Edições, 2002.
- [99] Andreas Tolk. Beyond Technical Interoperability - Introducing a Reference Model for Measures of Merit for Coalition Interoperability. In *8th International Command and Control Research and Technology Symposium*, 2003.
- [100] Darina Tothova and Beata Bellerova. Open Access to Information. In *EUNIS 2002*. EUNIS, FEUP Edições, 2002.
- [101] Dave Winer. RSS 2.0 Specification [online]. 2004 [cited 03-05-2004]. Available from: <http://blogs.law.harvard.edu/tech/rss>.
- [102] ASC X12. ASC X12 - Home [online]. 2003 [cited 15-03-2003]. Available from: <http://www.x12.org>.
- [103] ASC X12. X12A - Education Administration Purpose and Scope [online]. 2003 [cited 15-03-2003]. Available from: http://www.x12.org/x12org/subcommittees/sc_home.cfm?strSC=A.
- [104] J. Yang, W. J. van den Heuvel, and M. P. Papazoglou. Service Deployment for Virtual Enterprises. In *Proceedings of the workshop on Information technology for virtual enterprises*, pages 107–115. IEEE Computer Society, 2001.

Apêndice A

Estrutura da Base de Dados e Lista de Funcionalidades

A.1 Modelo Entidade-Relação

Na Figura A.1, apresenta-se o modelo entidade-relação da base de dados, construída para organizar as funcionalidades encontradas nos sistemas de informação universitários estudados. As principais entidades são: *feature* (funcionalidade), *category* (categoria), *keyword* (palavra-chave) e *system* (sistema).

Cada funcionalidade pode ser caracterizada por várias palavras-chave, bem como pertencer a múltiplas categorias. Por outro lado, a mesma funcionalidade pode existir em diferentes sistemas.

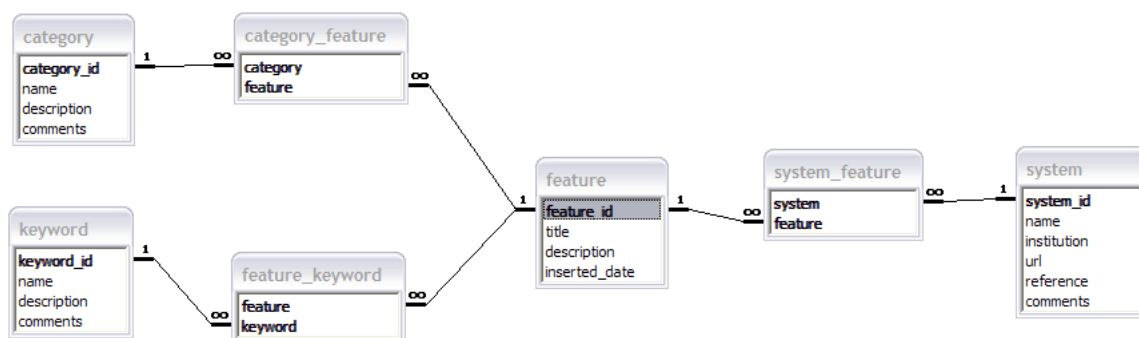


Figura A.1: Modelo Entidade-Relação

A.2 Lista de Funcionalidades

Na listagem seguinte, enumeram-se todas as funcionalidades encontradas nos sistemas de informação universitários estudados. A lista está organizada por ordem alfabética.

- Access to Bibliographic Search Database
- Allow Multiple Teachers by Course

- Allow students to define constraints in their schedule
- Allow teachers to customize grades parameters (components)
- Allow teachers to publish student's grades
- Alumni E-mail Forwarding
- Alumni's Record
- Approve Course Enrollment
- Approve Group Enrollment
- Approve Exam Enrollments
- Aquisitions Management
- Backups Management
- Budget Management
- Building's drawing
- Calendar: event invitations with accept/reject functionality
- Calendar: event reminders
- Calendar: Free/busy-time search
- Calendar: group several calendars into one
- Calendar: supports the iCAL standard
- Cards Management
- Cards Management: export data
- Career Advance Management
- Career Counseling
- Chat system associated with each course
- Chat system associated with each department
- Compute Degree's Final Average
- Content Personalization
- Copyright Protection Mechanisms
- Course catalog
- Course Enrollment Management
- Course Equivalences Management
- Course's bibliography
- Course's calendar
- Course's Complementary Page
- Course's coordinators can manage course's schedule
- Course's coordinators can request new time slots
- Course's Discussion Group
- Course's E-mail
- Course's Exams

- Course's Groups
- Course's Record
- Courses Search
- Course's statistical reports
- Course's students
- Course's students print with photos
- Course's summaries search
- Course's Teachers
- Customizable Interface
- Day or Month Based Calendar Browsing
- Decentralized content management
- Define deadlines for teachers to write the summaries
- Definition of rules to allow student's years transition
- Degree's Calendar
- Degree's Complementary Page
- Degree's Courses
- Degree's Discussion Group
- Degree's Record
- Degree's Search
- Degree's Statistical Information
- Degree's Students
- Degree's Teachers
- Digital Library
- Diploma Management
- Direct Connection to a Internet Publishing Area
- Disc Quota Management
- Disc Quotas Reloading
- Discussion Group Search
- Discussion groups management
- Discussion groups: users can create new groups online
- Document management using digital signatures
- Dorm Facilities Management
- E-Business system to manage purchases
- E-mail Alias Definition
- E-mail Auto-Response
- E-mail distribution system with recipients filtering and selection
- E-mail Forwarding

- E-mail Messaging
- Enrollment Scheduling Management
- Exam Enrollment Management
- Exam Enrollment Through the Web
- Export course's students list
- File sharing area associated with each course
- File sharing system associated with each department
- Financial Aid Processes Information
- Financial Management
- Forum associated with each course
- Forum associated with each department
- Full Search
- Group Enrollment Management
- Group's Discussion Group
- Groups Management for Document Distribution
- Guestbook management
- Help-desk System
- Institution's Calendar
- Integrate student's timetable/schedule with the published summaries
- Integrate Summaries with the Teacher's Schedule
- Integrated Webmail
- Integrates e-learning features
- Integration with an optical reader system to process documents
- Integration with Document Scanning System
- Integration with MS Word to produce reports/certificates using templates
- Inventory Management
- Job Center
- Job Center: indentify matches between the candidates and the offers
- Job Center: reminders
- Job Center: subscription
- Labour Agreements Management
- Library management
- Mailing list management
- Mailing lists: users can create new lists online
- Main Menu Customization
- Manage course's enrollment online
- Manage Course's Summaries

- Manage different schools and groups in the same institution
- Manage postgraduate thesis
- Manage student absences
- Manage teacher's absences
- Manage Teacher's Activity
- Manage teacher's extra-curricular activity
- Manage Thesis Workflow
- Manage Vacations
- Message Compilation Window
- Message exchange system associated with each course
- Message exchange system associated with each department
- Messaging System
- News Management
- News Management: Archive
- News Management: Archive
- News Management: Archive Search
- News Management: Categorized News
- News Management: Define Expiration Dates
- News Management: Define Headlines
- News Management: Personalization
- News Management: Priority Levels
- News Management: Search
- News Manager for Degrees, Courses, Groups, Departments
- Notices associated with each course
- Notices system associated with each department
- Open Source
- People search
- Personal Bookmark Manager
- Personal Calendar
- Personal Contact Manager
- Personal Grade History
- Personal Photo
- Personal Webpage
- Personalized E-mail Address
- Personalized Timetable
- Personalized Warnings
- Phone Directory

- Photo Gallery Management
- Postgraduation's Tuitions Management
- Print Quotas Management
- Print Quotas Reloading
- Project Application Management
- Publications Management
- Publications Management: users can modify the information.
- Publications Search
- Publish Tax Declarations
- R&D Project Search
- R&D Project's Complementary Page
- R&D Project's Record
- R&D Unit's Complementary Page
- R&D Unit's Record
- Recruitment Process Management
- Register for courses using the catalog
- Requests/Processes Management
- Researcher's Activities
- Researcher's Complementary Page
- Researcher's R&D Activities Management
- Researcher's Record
- Researchers Search
- Resources management
- Resources management: association between resources
- Resources management: associations between resources and the building's drawing
- Resources management: online information
- Resources management: reservation
- Resources management: statistical information
- Salary Management
- Search and filter course catalog
- Sincronization of information with mobile devices
- Single Login to the System
- SMS Messaging
- Staff's Record
- Statistical Information about Past Evaluations
- Student Association: e-mail address
- Student Associations: complementary page

- Student Associations: information
- Student's academic degrees
- Student's Applications Management
- Student's Applications Management: applicant automatic selection
- Student's Applications Management: applicant information
- Student's Applications Management: criteria definition
- Students can change their personal information in the system
- Student's Cash Account Management
- Student's complementary page
- Student's Courses
- Student's Financial Aid Management
- Student's Photo
- Student's Record
- Student's Tuitions Management
- Student's tutors can request more information to be filled
- Suggestion System
- Support common data formats for exchange with banks.
- Support pedagogical evaluations
- Support postgraduate studies
- Survey Management
- System: Access Control with SmartCards
- System: Accessibility Features
- System: Administration Guide
- System: Automatic Updates
- System: Bugtrack Management
- System: Change Password
- System: Contextual Help System
- System: Group Management
- System: Help System
- System: Log User Activity
- System: Manual
- System: Multilanguage Support
- System: Users Management
- System: Users's Permissions Management
- System: Users's Permissions Management By Area
- System: Users's Permissions Management By Feature
- System's Passwords Management

- Targeted Announcements
- Teachers can publish course's summaries
- Teacher's Complementary Page
- Teacher's Courses
- Teacher's Photo
- Teacher's Record
- Template Based, Personal Page
- Timetables Management: browse by subject (student, room, teacher, course, ...)
- Timetables Management: browse by time (day, week, year, ...)
- Timetables Management: Course's Timetable
- Timetables Management: Data Export
- Timetables Management: Group's Timetable
- Timetables Management: Room Validation
- Timetables Management: Room's Timetable
- Timetables Management: Student's Timetable
- Timetables Management: Teacher's Timetable
- Timetables Management: Teacher's timetable creation wizard
- Timetables Management: Timetable validation
- Treasury Management
- Trouble-Tickets
- Tuition Payment
- Unit's Portal
- User can Change Personal Information
- Verification and approval of grades
- View grades by course
- View grades by group
- View grades by student
- View student presences
- Web-based Administration
- Website Directory
- Workflow Management

Apêndice B

Especificação do Registo Académico do Aluno

B.1 Especificação do Elemento StudentRecord

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <xs:schema targetNamespace="http://schemas.up.pt/StudentRecord" xmlns="http://schemas.up.pt/StudentRecord" xmlns:xs="
  http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" attributeFormDefault="unqualified">
3
4   <!-- ##### -->
5   <!-- StudentRecord -->
6   <!-- ##### -->
7   <xs:element name="StudentRecord" type="StudentRecordType"/>
8
9   <!-- ##### -->
10  <!-- StudentRecordType -->
11  <!-- ##### -->
12  <xs:complexType name="StudentRecordType">
13    <xs:annotation>
14      <xs:documentation>
15        Registo do Aluno.
16      </xs:documentation>
17    </xs:annotation>
18    <xs:sequence>
19      <xs:element name="DocumentInfo" type="DocumentInfoType">
20        <xs:annotation>
21          <xs:documentation>
22            Informação sobre cada "instância" deste documento: data, emissor.
23          </xs:documentation>
24        </xs:annotation>
25      </xs:element>
26      <xs:element name="Student" type="StudentType">
27        <xs:annotation>
28          <xs:documentation>
29            Informação sobre o aluno.
30          </xs:documentation>
31        </xs:annotation>
32      </xs:element>
33      <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
34    </xs:sequence>
35  </xs:complexType>
36
37  <!-- ##### -->
38  <!-- DocumentInfoType -->
39  <!-- ##### -->
40  <xs:complexType name="DocumentInfoType">
41    <xs:annotation>
42      <xs:documentation>
43        Informação sobre o documento que representa o Registo do Aluno em XML.
44      </xs:documentation>
45    </xs:annotation>
46    <xs:sequence>
47      <xs:element name="DocumentCode" type="DocumentCodeType">
48        <xs:annotation>
49          <xs:documentation>
50            Identificador deste documento.
51          </xs:documentation>
52        </xs:annotation>
53      </xs:element>
54      <xs:element name="DocumentIssueDate" type="DocumentIssueDateType"/>
55      <xs:element name="Issuer" type="InstitutionType"/>
56      <xs:element name="StudentCode" type="StudentCodeType">
57        <xs:annotation>
```

```

58         <xs:documentation>
59             Identificador do aluno definido pela entidade emissora do documento.
60         </xs:documentation>
61     </xs:annotation>
62 </xs:element>
63 <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
64 </xs:sequence>
65 </xs:complexType>
66
67 <!-- DocumentCodeType -->
68 <xs:simpleType name="DocumentCodeType">
69     <xs:annotation>
70         <xs:documentation>
71             Código com significado local.
72             Atribuído pela entidade emissora do documento.
73         </xs:documentation>
74     </xs:annotation>
75     <xs:restriction base="xs:string"/>
76 </xs:simpleType>
77
78 <!-- DocumentIssueDateType -->
79 <xs:simpleType name="DocumentIssueDateType">
80     <xs:annotation>
81         <xs:documentation>
82             Data (dia e hora) de emissão do documento.
83         </xs:documentation>
84     </xs:annotation>
85     <xs:restriction base="xs:dateTime"/>
86 </xs:simpleType>
87
88 <!-- ##### -->
89 <!-- StudentType -->
90 <!-- ##### -->
91 <xs:complexType name="StudentType">
92     <xs:annotation>
93         <xs:documentation>
94             Registo de aluno.
95         </xs:documentation>
96     </xs:annotation>
97     <xs:sequence>
98         <xs:element name="Person" type="PersonType"/>
99         <xs:element name="Family" type="FamilyType" minOccurs="0"/>
100         <xs:element name="AdmissionExam" type="AdmissionExamType" minOccurs="0" maxOccurs="unbounded">
101             <xs:annotation>
102                 <xs:documentation>
103                     Provas de ingresso realizadas pelo estudante.
104                 </xs:documentation>
105             </xs:annotation>
106         </xs:element>
107         <xs:element name="AcademicDegree" type="AcademicDegreeType" minOccurs="0" maxOccurs="unbounded">
108             <xs:annotation>
109                 <xs:documentation>
110                     Graus académicos (Cursos) em que o aluno se inscreveu.
111                 </xs:documentation>
112             </xs:annotation>
113         </xs:element>
114         <xs:element name="HealthRecord" type="HealthRecordType" minOccurs="0">
115             <xs:annotation>
116                 <xs:documentation>
117                     Informação sobre os exames e imunizações (vacinas) do aluno.
118                 </xs:documentation>
119             </xs:annotation>
120         </xs:element>
121         <xs:element name="MilitaryRecord" type="MilitaryRecordType" minOccurs="0">
122             <xs:annotation>
123                 <xs:documentation>
124                     Informação militar.
125                 </xs:documentation>
126             </xs:annotation>
127         </xs:element>
128         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
129     </xs:sequence>
130 </xs:complexType>
131
132 <!-- ##### -->
133 <!-- PersonType -->
134 <!-- ##### -->
135 <xs:complexType name="PersonType">
136     <xs:annotation>
137         <xs:documentation>
138             Definição do tipo para representar uma pessoa.
139         </xs:documentation>
140     </xs:annotation>
141     <xs:sequence>
142         <xs:element name="IdentificationCode" type="IdentificationCodeType" minOccurs="0" maxOccurs="unbounded">
143             <xs:annotation>
144                 <xs:documentation>
145                     Uma pessoa pode ter vários códigos de identificação. Por exemplo, BI,
146                     NIF, etc.
147                 </xs:documentation>
148             </xs:annotation>
149         </xs:element>

```

```

149     <xs:element name="Name" type="NameType" />
150     <xs:element name="Birth" type="BirthType" minOccurs="0" />
151     <xs:element name="Deceased" type="DeceasedType" minOccurs="0">
152         <xs:annotation>
153             <xs:documentation>
154                 Este elemento existe se a pessoa já faleceu.
155             </xs:documentation>
156         </xs:annotation>
157     </xs:element>
158     <xs:element name="Gender" type="GenderType" minOccurs="0" />
159     <xs:element name="MaritalStatus" type="MaritalStatusType" minOccurs="0" />
160     <xs:element name="Citizenship" type="CitizenshipType" minOccurs="0" />
161     <xs:element name="Residency" type="AddressType" minOccurs="0">
162         <xs:annotation>
163             <xs:documentation>
164                 Residência "oficial", por isso só pode ter uma instância.
165             </xs:documentation>
166         </xs:annotation>
167     </xs:element>
168     <xs:element name="Contact" type="ContactType" minOccurs="0" maxOccurs="unbounded">
169         <xs:annotation>
170             <xs:documentation>
171                 Uma pessoa pode ter vários contactos de vários tipos. Por exemplo:
172                 telemóvel, telefone de casa, telefone do emprego, etc.
173             </xs:documentation>
174         </xs:annotation>
175     </xs:element>
176     <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
177 </xs:sequence>
178 </xs:complexType>
179
180 <!-- NameType -->
181 <xs:complexType name="NameType">
182     <xs:annotation>
183         <xs:documentation>
184             Nome de uma pessoa.
185         </xs:documentation>
186     </xs:annotation>
187     <xs:sequence>
188         <xs:element name="Title" type="NameTitleType" minOccurs="0" />
189         <xs:element name="FirstName" type="FirstNameType" minOccurs="0" />
190         <xs:element name="MiddleName" type="MiddleNameType" minOccurs="0" maxOccurs="unbounded" />
191         <xs:element name="LastName" type="LastNameType" minOccurs="0" />
192         <xs:element name="FullName" type="FullNameType" />
193         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
194     </xs:sequence>
195 </xs:complexType>
196
197 <!-- NameTitleType -->
198 <xs:simpleType name="NameTitleType">
199     <xs:restriction base="xs:string" />
200 </xs:simpleType>
201
202 <!-- FirstNameType -->
203 <xs:simpleType name="FirstNameType">
204     <xs:restriction base="xs:string" />
205 </xs:simpleType>
206
207 <!-- MiddleNameType -->
208 <xs:simpleType name="MiddleNameType">
209     <xs:restriction base="xs:string" />
210 </xs:simpleType>
211
212 <!-- LastNameType -->
213 <xs:simpleType name="LastNameType">
214     <xs:restriction base="xs:string" />
215 </xs:simpleType>
216
217 <!-- FullNameType -->
218 <xs:simpleType name="FullNameType">
219     <xs:restriction base="xs:string" />
220 </xs:simpleType>
221
222 <!-- BirthType -->
223 <xs:complexType name="BirthType">
224     <xs:sequence>
225         <xs:element name="BirthDate" type="xs:date" />
226         <xs:element name="BirthCity" type="CityType" minOccurs="0" />
227         <xs:element name="BirthCountry" type="CountryType" minOccurs="0" />
228         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
229     </xs:sequence>
230 </xs:complexType>
231
232 <!-- DeceasedType -->
233 <xs:complexType name="DeceasedType">
234     <xs:sequence>
235         <xs:element name="DeceasedIndicator" type="xs:boolean">
236             <xs:annotation>
237                 <xs:documentation>
238                     Verdadeiro se já faleceu.
239                 </xs:documentation>
240             </xs:annotation>
241         </xs:element>
242         <xs:element name="DeceasedDate" type="xs:date" minOccurs="0" />

```

```

242         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
243     </xs:sequence>
244 </xs:complexType>
245
246 <!-- GenderType -->
247 <xs:simpleType name="GenderType">
248     <xs:annotation>
249         <xs:documentation>Gênero da pessoa. Não preenchido em caso de ser desconhecido.</
                xs:documentation>
250     </xs:annotation>
251     <xs:restriction base="xs:string">
252         <xs:enumeration value="F">
253             <xs:annotation>
254                 <xs:documentation>Female / Feminino.</xs:documentation>
255             </xs:annotation>
256         </xs:enumeration>
257         <xs:enumeration value="M">
258             <xs:annotation>
259                 <xs:documentation>Male / Masculino.</xs:documentation>
260             </xs:annotation>
261         </xs:enumeration>
262         <xs:enumeration value="U">
263             <xs:annotation>
264                 <xs:documentation>Unspecified / Não especificado.</xs:documentation>
265             </xs:annotation>
266         </xs:enumeration>
267     </xs:restriction>
268 </xs:simpleType>
269
270 <!-- MaritalStatus -->
271 <xs:simpleType name="MaritalStatusType">
272     <xs:annotation>
273         <xs:documentation>Estado civil da pessoa.</xs:documentation>
274     </xs:annotation>
275     <xs:restriction base="xs:string">
276         <xs:enumeration value="S">
277             <xs:annotation>
278                 <xs:documentation>Single / Solteiro.</xs:documentation>
279             </xs:annotation>
280         </xs:enumeration>
281         <xs:enumeration value="M">
282             <xs:annotation>
283                 <xs:documentation>Married / Casado.</xs:documentation>
284             </xs:annotation>
285         </xs:enumeration>
286         <xs:enumeration value="D">
287             <xs:annotation>
288                 <xs:documentation>Divorced / Divorciado.</xs:documentation>
289             </xs:annotation>
290         </xs:enumeration>
291         <xs:enumeration value="W">
292             <xs:annotation>
293                 <xs:documentation>Widowed / Viúvo.</xs:documentation>
294             </xs:annotation>
295         </xs:enumeration>
296         <xs:enumeration value="U">
297             <xs:annotation>
298                 <xs:documentation>Unspecified / Não especificado.</xs:documentation>
299             </xs:annotation>
300         </xs:enumeration>
301     </xs:restriction>
302 </xs:simpleType>
303
304 <!-- CitizenshipType -->
305 <xs:simpleType name="CitizenshipType">
306     <xs:restriction base="CountryType"/>
307 </xs:simpleType>
308
309 <!-- IdentificationCodeType -->
310 <xs:complexType name="IdentificationCodeType">
311     <xs:annotation>
312         <xs:documentation>
313             Definição do tipo para representar números de identificação de cartões. Por exemplo:
314             bilhete de identidade, cartão de contribuinte, etc.
315         </xs:documentation>
316     </xs:annotation>
317     <xs:sequence>
318         <xs:element name="IdentificationCodeCode" type="IdentificationCodeCodeType" minOccurs="0">
319             <xs:annotation>
320                 <xs:documentation>
321                     Código identificativo do sistema de identificação usado.
322                     Usar nos casos em que o sistema de identificação é um dos tipos pré-
323                     definidos.
324                 </xs:documentation>
325             </xs:annotation>
326         </xs:element>
327         <xs:element name="IdentificationCodeDescription" type="IdentificationCodeDescriptionType"/>
328         <xs:element name="IdentificationCodeValue" type="IdentificationCodeValueType"/>
329         <xs:element name="IdentificationCodeIssuer" type="IdentificationCodeIssuerType" minOccurs="0"/>
330         <xs:element name="IdentificationCodeIssueDate" type="IdentificationCodeIssueDateType"
331             minOccurs="0"/>
332         <xs:element name="IdentificationCodeExpirationDate" type="IdentificationCodeExpirationDateType"
333             minOccurs="0"/>

```

```

330         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
331     </xs:sequence>
332 </xs:complexType>
333
334 <!-- IdentificationCodeCodeType -->
335 <xs:simpleType name="IdentificationCodeCodeType">
336     <xs:annotation>
337         <xs:documentation>
338             Código identificador do sistema de identificação usado. Usar nos casos discriminados.
339         </xs:documentation>
340     </xs:annotation>
341     <xs:restriction base="xs:string">
342         <xs:enumeration value="BI">
343             <xs:annotation>
344                 <xs:documentation>Bilhete de Identidade.</xs:documentation>
345             </xs:annotation>
346         </xs:enumeration>
347         <xs:enumeration value="NIF">
348             <xs:annotation>
349                 <xs:documentation>Número de Identificação Fiscal.</xs:documentation>
350             </xs:annotation>
351         </xs:enumeration>
352         <xs:enumeration value="NCC">
353             <xs:annotation>
354                 <xs:documentation>Número da Carta de Condução.</xs:documentation>
355             </xs:annotation>
356         </xs:enumeration>
357         <xs:enumeration value="NSS">
358             <xs:annotation>
359                 <xs:documentation>Número da Segurança Social.</xs:documentation>
360             </xs:annotation>
361         </xs:enumeration>
362         <xs:enumeration value="NP">
363             <xs:annotation>
364                 <xs:documentation>Número do Passaporte.</xs:documentation>
365             </xs:annotation>
366         </xs:enumeration>
367     </xs:restriction>
368 </xs:simpleType>
369
370 <!-- IdentificationCodeDescriptionType -->
371 <xs:simpleType name="IdentificationCodeDescriptionType">
372     <xs:restriction base="xs:string"/>
373 </xs:simpleType>
374
375 <!-- IdentificationCodeValueType -->
376 <xs:simpleType name="IdentificationCodeValueType">
377     <xs:annotation>
378         <xs:documentation>
379             Código segundo este sistema de identificação.
380             Por exemplo, se estivermos a armazenar o número de BI, incluir aqui o próprio número.
381         </xs:documentation>
382     </xs:annotation>
383     <xs:restriction base="xs:string"/>
384 </xs:simpleType>
385
386 <!-- IdentificationCodeIssuerType -->
387 <xs:simpleType name="IdentificationCodeIssuerType">
388     <xs:annotation>
389         <xs:documentation>
390             Informação sobre o emissor do código de identificação. Por exemplo: Estado (BI), DGV (
391             carta de condução).
392         </xs:documentation>
393     </xs:annotation>
394     <xs:restriction base="xs:string"/>
395 </xs:simpleType>
396
397 <!-- IdentificationCodeIssueDateType -->
398 <xs:simpleType name="IdentificationCodeIssueDateType">
399     <xs:restriction base="xs:date"/>
400 </xs:simpleType>
401
402 <!-- IdentificationCodeExpirationDateType -->
403 <xs:simpleType name="IdentificationCodeExpirationDateType">
404     <xs:restriction base="xs:date"/>
405 </xs:simpleType>
406
407 <!-- ##### -->
408 <!-- ContactType -->
409 <!-- ##### -->
410 <xs:complexType name="ContactType">
411     <xs:sequence>
412         <xs:element name="ContactDescription" type="ContactDescriptionType"/>
413         <xs:choice>
414             <xs:element name="Address" type="AddressType"/>
415             <xs:element name="Phone" type="PhoneType"/>
416             <xs:element name="Email" type="EmailType"/>
417             <xs:element name="URL" type="URLType"/>
418         </xs:choice>
419         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
420     </xs:sequence>
421 </xs:complexType>
422 <!-- ContactDescriptionType -->

```

```

423 <xs:simpleType name="ContactDescriptionType">
424   <xs:restriction base="xs:string" />
425 </xs:simpleType>
426
427 <!-- AddressType -->
428 <xs:complexType name="AddressType">
429   <xs:sequence>
430     <xs:element name="AddressAttentionLine" type="AddressAttentionLineType" minOccurs="0"
431       maxOccurs="unbounded" />
432     <xs:element name="AddressLine" type="AddressLineType" maxOccurs="unbounded" />
433     <xs:element name="AddressPostalCode" type="PostalCodeType" />
434     <xs:element name="AddressCity" type="CityType" />
435     <xs:element name="AddressCountry" type="CountryType" />
436     <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
437   </xs:sequence>
438 </xs:complexType>
439
440 <!-- PhoneType -->
441 <xs:complexType name="PhoneType">
442   <xs:sequence>
443     <xs:element name="PhoneCountryCode" type="PhoneCountryCodeType" minOccurs="0" />
444     <xs:element name="PhoneNumber" type="PhoneNumberType" />
445     <xs:element name="PhoneNumberExtension" type="PhoneNumberExtensionType" minOccurs="0" />
446   </xs:sequence>
447 </xs:complexType>
448
449 <!-- EmailType -->
450 <xs:simpleType name="EmailType">
451   <xs:restriction base="xs:string">
452     <xs:pattern value="\w@\w.\w"/>
453   </xs:restriction>
454 </xs:simpleType>
455
456 <!-- URLType -->
457 <xs:simpleType name="URLType">
458   <xs:restriction base="xs:anyURI" />
459 </xs:simpleType>
460
461 <!-- AddressAttentionLineType -->
462 <xs:simpleType name="AddressAttentionLineType">
463   <xs:restriction base="xs:string" />
464 </xs:simpleType>
465
466 <!-- AddressLineType -->
467 <xs:simpleType name="AddressLineType">
468   <xs:restriction base="xs:string" />
469 </xs:simpleType>
470
471 <!-- AddressPostalCodeType -->
472 <xs:simpleType name="PostalCodeType">
473   <xs:restriction base="xs:string" />
474 </xs:simpleType>
475
476 <!-- AddressCityType -->
477 <xs:simpleType name="CityType">
478   <xs:restriction base="xs:string" />
479 </xs:simpleType>
480
481 <!-- CountryType -->
482 <xs:simpleType name="CountryType">
483   <xs:annotation>
484     <xs:documentation>
485       Identificador de um país usando o código numérico da norma ISO 3166-1. Exemplo:
486       Portugal = 620.
487     </xs:documentation>
488   </xs:annotation>
489   <xs:restriction base="xs:string">
490     <xs:pattern value="\d\d\d"/>
491   </xs:restriction>
492 </xs:simpleType>
493
494 <!-- PhoneCountryCode -->
495 <xs:simpleType name="PhoneCountryCodeType">
496   <xs:restriction base="xs:string" />
497 </xs:simpleType>
498
499 <!-- PhoneNumber -->
500 <xs:simpleType name="PhoneNumberType">
501   <xs:restriction base="xs:string" />
502 </xs:simpleType>
503
504 <!-- PhoneNumberExtension -->
505 <xs:simpleType name="PhoneNumberExtensionType">
506   <xs:restriction base="xs:string" />
507 </xs:simpleType>
508
509 <!-- ##### -->
510 <!-- FamilyType -->
511 <!-- ##### -->
512 <xs:complexType name="FamilyType">
513   <xs:annotation>
514     <xs:documentation>
515       Representa a filiação de uma pessoa. A mãe e o pai são opcionais porque pode não haver
516       informação sobre os dois.

```



```

514         </xs:documentation>
515     </xs:annotation>
516     <xs:sequence>
517         <xs:element name="Mother" type="PersonType" minOccurs="0"/>
518         <xs:element name="Father" type="PersonType" minOccurs="0"/>
519         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
520     </xs:sequence>
521 </xs:complexType>
522
523 <!-- NoteType -->
524 <xs:simpleType name="NoteType">
525     <xs:annotation>
526         <xs:documentation>
527             Tipo que representa um elemento genérico para a definição de anotações.
528         </xs:documentation>
529     </xs:annotation>
530     <xs:restriction base="xs:string"/>
531 </xs:simpleType>
532
533 <!-- ##### -->
534 <!-- AdmissionExamType -->
535 <!-- ##### -->
536 <xs:complexType name="AdmissionExamType">
537     <xs:annotation>
538         <xs:documentation>
539             Prova de ingresso.
540         </xs:documentation>
541     </xs:annotation>
542     <xs:sequence>
543         <xs:element name="AdmissionExamName" type="AdmissionExamNameType" minOccurs="0"/>
544         <xs:element name="AdmissionExamCode" type="AdmissionExamCodeType" minOccurs="0"/>
545         <xs:element name="AdmissionExamDate" type="AdmissionExamDateType" minOccurs="0"/>
546         <xs:element name="AdmissionExamGrade" type="AcademicGradeType" minOccurs="0"/>
547         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
548     </xs:sequence>
549 </xs:complexType>
550
551 <!-- AdmissionExamTitleType -->
552 <xs:simpleType name="AdmissionExamNameType">
553     <xs:annotation>
554         <xs:documentation>
555             Nome da prova de admissão. Exemplo: "Prova_específica_de_Matemática", "GMAT".
556         </xs:documentation>
557     </xs:annotation>
558     <xs:restriction base="xs:string"/>
559 </xs:simpleType>
560
561 <!-- AdmissionExamLevelType -->
562 <xs:simpleType name="AdmissionExamCodeType">
563     <xs:annotation>
564         <xs:documentation>
565             Código identificador da prova de admissão.
566         </xs:documentation>
567     </xs:annotation>
568     <xs:restriction base="xs:string"/>
569 </xs:simpleType>
570
571 <!-- AdmissionExamDateType -->
572 <xs:simpleType name="AdmissionExamDateType">
573     <xs:annotation>
574         <xs:documentation>
575             Data da prova de admissão.
576         </xs:documentation>
577     </xs:annotation>
578     <xs:restriction base="xs:date"/>
579 </xs:simpleType>
580
581 <!-- ##### -->
582 <!-- AcademicDegreeType -->
583 <!-- ##### -->
584 <xs:complexType name="AcademicDegreeType">
585     <xs:annotation>
586         <xs:documentation>
587             Matrícula de um aluno num grau académico de uma instituição.
588             Pode não estar terminado.
589             Por exemplo: se um aluno faz uma licenciatura e depois um mestrado tem dois elementos
590             destes.
591         </xs:documentation>
592     </xs:annotation>
593     <xs:sequence>
594         <xs:element name="Degree" type="DegreeType">
595             <xs:annotation>
596                 <xs:documentation>
597                     Identificação do curso.
598                 </xs:documentation>
599             </xs:annotation>
600         </xs:element>
601         <xs:element name="StudentCode" type="StudentCodeType">
602             <xs:annotation>
603                 <xs:documentation>
604                     Identificador do aluno no contexto deste grau académico.
605                     Este código é atribuído pela instituição responsável pelo grau
606                     académico.
607                 </xs:documentation>

```

```

606         </xs:annotation>
607     </xs:element>
608     <xs:element name="DegreeAdmissionDate" type="xs:date" />
609     <xs:element name="DegreeStudentStatus" type="DegreeStudentStatusType" minOccurs="0" />
610     <xs:element name="DegreeConclusionDate" type="xs:date" minOccurs="0" />
611     <xs:element name="DegreeAcademicGradeAverage" type="AcademicGradeType" minOccurs="0">
612         <xs:annotation>
613             <xs:documentation>
614                 No caso do curso estar concluído, a média final.
615             </xs:documentation>
616         </xs:annotation>
617     </xs:element>
618     <xs:element name="DegreeECTSGradeAverage" type="ECTSGradeType" minOccurs="0">
619         <xs:annotation>
620             <xs:documentation>
621                 Média final na escala ECTS.
622             </xs:documentation>
623         </xs:annotation>
624     </xs:element>
625     <xs:element name="AcademicSession" type="AcademicSessionType" minOccurs="0" maxOccurs="unbounded">
626         <xs:annotation>
627             <xs:documentation>
628                 Um período lectivo.
629                 Um curso tem associados vários períodos lectivos. Normalmente "anos_lectivos".
630             </xs:documentation>
631         </xs:annotation>
632     </xs:element>
633     <xs:element name="AcademicAward" type="AcademicAwardType" minOccurs="0" maxOccurs="unbounded">
634         <xs:annotation>
635             <xs:documentation>
636                 Referência aos prêmios académicos.
637             </xs:documentation>
638         </xs:annotation>
639     </xs:element>
640     <xs:element name="ExtraActivity" type="ExtraActivityType" minOccurs="0" maxOccurs="unbounded">
641         <xs:annotation>
642             <xs:documentation>
643                 Referência a actividades extra-curriculares de relevância.
644             </xs:documentation>
645         </xs:annotation>
646     </xs:element>
647     <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
648 </xs:sequence>
649 </xs:complexType>
650
651 <!-- ##### -->
652 <!-- StudentCodeType -->
653 <!-- ##### -->
654 <xs:simpleType name="StudentCodeType">
655     <xs:annotation>
656         <xs:documentation>
657             Definição do tipo para representar o identificador de um aluno num curso.
658             Cada instituição pode usar um sistema diferente.
659         </xs:documentation>
660     </xs:annotation>
661     <xs:restriction base="xs:string" />
662 </xs:simpleType>
663
664 <!-- ##### -->
665 <!-- StudentStatusType -->
666 <!-- ##### -->
667 <xs:simpleType name="DegreeStudentStatusType">
668     <xs:annotation>
669         <xs:documentation>
670             Definição do tipo para representar o estado de um aluno numa matrícula.
671         </xs:documentation>
672     </xs:annotation>
673     <xs:restriction base="xs:string">
674         <xs:enumeration value="AC">
675             <xs:annotation>
676                 <xs:documentation>Mudança de Curso / Academic Degree Change.</xs:documentation>
677             </xs:annotation>
678         </xs:enumeration>
679         <xs:enumeration value="CS">
680             <xs:annotation>
681                 <xs:documentation>Concluiu Parte Escolar / Concluded Scholar Component.</xs:documentation>
682             </xs:annotation>
683         </xs:enumeration>
684         <xs:enumeration value="CN">
685             <xs:annotation>
686                 <xs:documentation>Concluído / Concluded.</xs:documentation>
687             </xs:annotation>
688         </xs:enumeration>
689         <xs:enumeration value="DC">
690             <xs:annotation>
691                 <xs:documentation>Falecido / Deceased.</xs:documentation>
692             </xs:annotation>
693         </xs:enumeration>
694         <xs:enumeration value="DR">
695             <xs:annotation>

```

```

696         <xs:documentation>Desistiu / Dropped.</xs:documentation>
697     </xs:annotation>
698 </xs:enumeration>
699 <xs:enumeration value="EA">
700     <xs:annotation>
701         <xs:documentation>Anulação da Matrícula / Enrollment Annulled.</
              xs:documentation>
702     </xs:annotation>
703 </xs:enumeration>
704 <xs:enumeration value="ES">
705     <xs:annotation>
706         <xs:documentation>Suspensão da Matrícula / Enrollment Suspended.</
              xs:documentation>
707     </xs:annotation>
708 </xs:enumeration>
709 <xs:enumeration value="FQ">
710     <xs:annotation>
711         <xs:documentation>A frequentar / Frequenting.</xs:documentation>
712     </xs:annotation>
713 </xs:enumeration>
714 <xs:enumeration value="IN">
715     <xs:annotation>
716         <xs:documentation>Interrompido / Interrupted.</xs:documentation>
717     </xs:annotation>
718 </xs:enumeration>
719 <xs:enumeration value="NE">
720     <xs:annotation>
721         <xs:documentation>Não Inscrito / Not Enrolled.</xs:documentation>
722     </xs:annotation>
723 </xs:enumeration>
724 <xs:enumeration value="NP">
725     <xs:annotation>
726         <xs:documentation>Não Pagou Propina / Fee Not Paid.</xs:documentation>
727     </xs:annotation>
728 </xs:enumeration>
729 <xs:enumeration value="PE">
730     <xs:annotation>
731         <xs:documentation>Inscrição Provisória / Provisory Enrollment.</
              xs:documentation>
732     </xs:annotation>
733 </xs:enumeration>
734 <xs:enumeration value="PM">
735     <xs:annotation>
736         <xs:documentation>Permutado / Permuted.</xs:documentation>
737     </xs:annotation>
738 </xs:enumeration>
739 <xs:enumeration value="PR">
740     <xs:annotation>
741         <xs:documentation>Prescrição / Prescribed.</xs:documentation>
742     </xs:annotation>
743 </xs:enumeration>
744 <xs:enumeration value="RA">
745     <xs:annotation>
746         <xs:documentation>Recolocado Noutro Curso / Repositioned in Other Academic
              Degree.</xs:documentation>
747     </xs:annotation>
748 </xs:enumeration>
749 <xs:enumeration value="RE">
750     <xs:annotation>
751         <xs:documentation>Reingresso / Reenrollment.</xs:documentation>
752     </xs:annotation>
753 </xs:enumeration>
754 <xs:enumeration value="TR">
755     <xs:annotation>
756         <xs:documentation>Transferência / Transference.</xs:documentation>
757     </xs:annotation>
758 </xs:enumeration>
759 <xs:enumeration value="UK">
760     <xs:annotation>
761         <xs:documentation>Desconhecido / Unknown.</xs:documentation>
762     </xs:annotation>
763 </xs:enumeration>
764 </xs:restriction>
765 </xs:simpleType>
766
767 <!-- ##### -->
768 <!-- InstitutionType -->
769 <!-- ##### -->
770 <xs:complexType name="InstitutionType">
771     <xs:sequence>
772         <xs:element name="InstitutionCode" type="InstitutionCodeType">
773             <xs:annotation>
774                 <xs:documentation>
775                     Código identificador da instituição no contexto nacional.
776                     Em Portugal, o código definido pelo Ministério da Ciência e Ensino
777                     Superior. Exemplo: FEUP - 1105.
778                 </xs:documentation>
779             </xs:annotation>
780         </xs:element>
781         <xs:element name="InstitutionName" type="InstitutionNameType"/>
782         <xs:element name="InstitutionLevel" type="InstitutionLevelType" minOccurs="0"/>
783         <xs:element name="InstitutionCountry" type="CountryType"/>
784         <xs:element name="Contact" type="ContactType" minOccurs="0" maxOccurs="unbounded"/>
785         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>

```

```

785         </xs:sequence>
786     </xs:complexType>
787
788     <!-- InstitutionCodeType -->
789     <xs:simpleType name="InstitutionCodeType">
790         <xs:annotation>
791             <xs:documentation>
792                 Código identificador da instituição num contexto nacional.
793             </xs:documentation>
794         </xs:annotation>
795         <xs:restriction base="xs:string"/>
796     </xs:simpleType>
797
798     <!-- InstitutionNameType -->
799     <xs:simpleType name="InstitutionNameType">
800         <xs:restriction base="xs:string"/>
801     </xs:simpleType>
802
803     <!-- InstitutionLevelType -->
804     <xs:simpleType name="InstitutionLevelType">
805         <xs:annotation>
806             <xs:documentation>
807                 Nível académico da instituição: universitário, politécnico.
808             </xs:documentation>
809         </xs:annotation>
810         <xs:restriction base="xs:string">
811             <xs:enumeration value="UL">
812                 <xs:annotation>
813                     <xs:documentation>Nível Universitário.</xs:documentation>
814                 </xs:annotation>
815             </xs:enumeration>
816             <xs:enumeration value="PL">
817                 <xs:annotation>
818                     <xs:documentation>Nível Politécnico.</xs:documentation>
819                 </xs:annotation>
820             </xs:enumeration>
821         </xs:restriction>
822     </xs:simpleType>
823
824     <!-- ##### -->
825     <!-- DegreeType -->
826     <!-- ##### -->
827     <xs:complexType name="DegreeType">
828         <xs:annotation>
829             <xs:documentation>
830                 Grau académico. Por exemplo: licenciatura, mestrado.
831             </xs:documentation>
832         </xs:annotation>
833         <xs:sequence>
834             <xs:element name="Institution" type="InstitutionType">
835                 <xs:annotation>
836                     <xs:documentation>
837                         Instituição responsável por este curso.
838                     </xs:documentation>
839                 </xs:annotation>
840             </xs:element>
841             <xs:element name="DegreeCode" type="DegreeCodeType" minOccurs="0">
842                 <xs:annotation>
843                     <xs:documentation>
844                         Código identificador do curso no contexto da instituição responsável
845                         por ele.
846                     </xs:documentation>
847                 </xs:annotation>
848             </xs:element>
849             <xs:element name="DegreeName" type="xs:string">
850                 <xs:annotation>
851                     <xs:documentation>
852                         Nome do curso.
853                     </xs:documentation>
854                 </xs:annotation>
855             </xs:element>
856             <xs:element name="CurricularProgram" type="CurricularProgramType" minOccurs="0">
857                 <xs:annotation>
858                     <xs:documentation>
859                         Plano de estudos.
860                     </xs:documentation>
861                 </xs:annotation>
862             </xs:element>
863             <xs:element name="DegreeLevel" type="DegreeLevelType" minOccurs="0">
864                 <xs:annotation>
865                     <xs:documentation>
866                         Grau académico oferecido pelo curso.
867                     </xs:documentation>
868                 </xs:annotation>
869             </xs:element>
870             <xs:element name="DegreeDuration" type="xs:duration" minOccurs="0">
871                 <xs:annotation>
872                     <xs:documentation>
873                         Duração do curso.
874                     </xs:documentation>
875                 </xs:annotation>
876             </xs:element>
877             <xs:element name="Contact" type="ContactType" minOccurs="0" maxOccurs="unbounded">
878                 <xs:annotation>

```

```

878         <xs:documentation>
879             Contactos do curso. Por exemplo: url, direcção, secretaria, etc.
880         </xs:documentation>
881     </xs:annotation>
882 </xs:element>
883 <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
884 </xs:sequence>
885 </xs:complexType>
886
887 <!-- CurricularProgramType -->
888 <xs:complexType name="CurricularProgramType">
889     <xs:annotation>
890         <xs:documentation>
891             Referência para um plano curricular.
892         </xs:documentation>
893     </xs:annotation>
894     <xs:sequence>
895         <xs:element name="CurricularProgramCode" type="CurricularProgramCodeType" />
896         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
897     </xs:sequence>
898 </xs:complexType>
899
900 <!-- CurricularProgramCodeType -->
901 <xs:simpleType name="CurricularProgramCodeType">
902     <xs:annotation>
903         <xs:documentation>
904             Código identificador do plano de estudos no contexto da instituição responsável pelo
905             curso.
906         </xs:documentation>
907     </xs:annotation>
908     <xs:restriction base="xs:string" />
909 </xs:simpleType>
910
911 <!-- DegreeCodeType -->
912 <xs:simpleType name="DegreeCodeType">
913     <xs:annotation>
914         <xs:documentation>
915             Identificador de um curso, definido pela entidade responsável pelo curso.
916         </xs:documentation>
917     </xs:annotation>
918     <xs:restriction base="xs:string" />
919 </xs:simpleType>
920
921 <!-- DegreeLevelType -->
922 <xs:simpleType name="DegreeLevelType">
923     <xs:annotation>
924         <xs:documentation>
925             Grau académico oferecido pelo curso: bacharelamento, licenciatura, pós-graduação,
926             mestrado ou doutoramento.
927         </xs:documentation>
928     </xs:annotation>
929     <xs:restriction base="xs:string">
930         <xs:enumeration value="BD">
931             <xs:annotation>
932                 <xs:documentation> Bacharelamento. </xs:documentation>
933             </xs:annotation>
934         </xs:enumeration>
935         <xs:enumeration value="GD">
936             <xs:annotation>
937                 <xs:documentation> Licenciatura (Graduação). </xs:documentation>
938             </xs:annotation>
939         </xs:enumeration>
940         <xs:enumeration value="PD">
941             <xs:annotation>
942                 <xs:documentation> Pós-Graduação. </xs:documentation>
943             </xs:annotation>
944         </xs:enumeration>
945         <xs:enumeration value="MD">
946             <xs:annotation>
947                 <xs:documentation> Mestrado. </xs:documentation>
948             </xs:annotation>
949         </xs:enumeration>
950         <xs:enumeration value="DD">
951             <xs:annotation>
952                 <xs:documentation> Doutoramento. </xs:documentation>
953             </xs:annotation>
954         </xs:enumeration>
955     </xs:restriction>
956 </xs:simpleType>
957
958 <!-- AcademicSessionType -->
959 <xs:complexType name="AcademicSessionType">
960     <xs:annotation>
961         <xs:documentation>
962             Representação de uma sessão académica, matrícula ou inscrição.
963         </xs:documentation>
964     </xs:annotation>
965     <xs:sequence>
966         <xs:element name="AcademicSessionStartDate" type="xs:date" minOccurs="0" />
967         <xs:element name="AcademicSessionEndDate" type="xs:date" minOccurs="0" />
968         <xs:element name="Course" type="CourseType" minOccurs="0" maxOccurs="unbounded">
969             <xs:annotation>
970                 <xs:documentation>
971                     Inscrições feitas pelo aluno durante este período.

```

```

970      Cada inscrição corresponde a uma disciplina.
971      </xs:documentation>
972    </xs:annotation>
973  </xs:element>
974  <xs:element name="Thesis" type="ThesisType" minOccurs="0">
975    <xs:annotation>
976      <xs:documentation>
977        No caso de aluno de mestrado ou doutoramento, representa informação
          sobre a tese.
978      </xs:documentation>
979    </xs:annotation>
980  </xs:element>
981  <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
982  </xs:sequence>
983 </xs:complexType>
984
985 <!-- ##### -->
986 <!-- CourseType -->
987 <!-- ##### -->
988 <xs:complexType name="CourseType">
989   <xs:annotation>
990     <xs:documentation>
991       Disciplina.
992     </xs:documentation>
993   </xs:annotation>
994   <xs:sequence>
995     <xs:element name="CourseCode" type="CourseCodeType" minOccurs="0"/>
996     <xs:element name="CourseName" type="xs:string"/>
997     <xs:element name="CourseStartDate" type="xs:date" minOccurs="0"/>
998     <xs:element name="CourseEndDate" type="xs:date" minOccurs="0"/>
999     <xs:element name="CourseCreditUnits" type="CourseCreditUnitsType" minOccurs="0">
1000       <xs:annotation>
1001         <xs:documentation>
1002           Unidades de crédito desta disciplina segundo o sistema da instituição.
1003         </xs:documentation>
1004       </xs:annotation>
1005     </xs:element>
1006     <xs:element name="CourseECTSCreditUnits" type="ECTSCreditUnitsType" minOccurs="0">
1007       <xs:annotation>
1008         <xs:documentation>
1009           Unidades de crédito desta disciplina segundo o sistema ECTS.
1010         </xs:documentation>
1011       </xs:annotation>
1012     </xs:element>
1013     <xs:element name="CourseClassSize" type="xs:integer" minOccurs="0">
1014       <xs:annotation>
1015         <xs:documentation>
1016           Número de alunos inscritos na disciplina.
1017         </xs:documentation>
1018       </xs:annotation>
1019     </xs:element>
1020     <xs:element name="CourseAcademicGrade" type="AcademicGradeType" minOccurs="0">
1021       <xs:annotation>
1022         <xs:documentation>
1023           Classificação final do aluno.
1024         </xs:documentation>
1025       </xs:annotation>
1026     </xs:element>
1027     <xs:element name="CourseECTSGrade" type="ECTSGradeType" minOccurs="0">
1028       <xs:annotation>
1029         <xs:documentation>
1030           Classificação final na escala ECTS.
1031         </xs:documentation>
1032       </xs:annotation>
1033     </xs:element>
1034     <xs:element name="CourseEquivalenceInformation" type="EquivalenceInformationType" minOccurs="0"
1035       maxOccurs="unbounded">
1036       <xs:annotation>
1037         <xs:documentation>
1038           Este elemento vai existir no caso desta disciplina ter sido feita por
1039             equivalência de uma outra.
1040           Permite ter várias disciplinas a "contribuir" para esta.
1041         </xs:documentation>
1042       </xs:annotation>
1043     </xs:element>
1044     <xs:element name="Contact" type="ContactType" minOccurs="0" maxOccurs="unbounded">
1045       <xs:annotation>
1046         <xs:documentation>
1047           Contactos relativos à disciplina. Por exemplo: regente, página web.
1048         </xs:documentation>
1049       </xs:annotation>
1050     </xs:element>
1051     <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
1052   </xs:sequence>
1053 </xs:complexType>
1054
1055 <!-- CourseCodeType -->
1056 <xs:simpleType name="CourseCodeType">
1057   <xs:annotation>
1058     <xs:documentation>
1059       Código identificador da disciplina, atribuído pela instituição reponsável pela
1060       disciplina.
1061     </xs:documentation>
1062   </xs:annotation>

```

```

1060         <xs:restriction base="xs:string"/>
1061     </xs:simpleType>
1062
1063     <!-- CourseCreditUnits -->
1064     <xs:simpleType name="CourseCreditUnitsType">
1065         <xs:restriction base="xs:decimal"/>
1066     </xs:simpleType>
1067
1068     <!-- ECTSCreditUnitsType -->
1069     <xs:simpleType name="ECTSCreditUnitsType">
1070         <xs:annotation>
1071             <xs:documentation>
1072                 European Credit Transfer System.
1073             </xs:documentation>
1074         </xs:annotation>
1075         <xs:restriction base="xs:decimal"/>
1076     </xs:simpleType>
1077
1078     <!-- GradeType -->
1079     <xs:simpleType name="AcademicGradeType">
1080         <xs:annotation>
1081             <xs:documentation>
1082                 Classificação segundo o sistema local.
1083                 Por exemplo: 0..20; A..F.
1084             </xs:documentation>
1085         </xs:annotation>
1086         <xs:restriction base="xs:string"/>
1087     </xs:simpleType>
1088
1089     <!-- ECTSGradeType -->
1090     <xs:simpleType name="ECTSGradeType">
1091         <xs:annotation>
1092             <xs:documentation>
1093                 Classificação segundo o sistema ECTS.
1094             </xs:documentation>
1095         </xs:annotation>
1096         <xs:restriction base="xs:string">
1097             <xs:enumeration value="A">
1098                 <xs:annotation>
1099                     <xs:documentation>
1100                         10% dos alunos normalmente atingem este valor.
1101                         EXCELENTE - desempenho único, com erros pontuais.
1102                     </xs:documentation>
1103                 </xs:annotation>
1104             </xs:enumeration>
1105             <xs:enumeration value="B">
1106                 <xs:annotation>
1107                     <xs:documentation>
1108                         25% dos alunos normalmente atingem este valor.
1109                         MUITO BOM - acima da média, mas com alguns erros.
1110                     </xs:documentation>
1111                 </xs:annotation>
1112             </xs:enumeration>
1113             <xs:enumeration value="C">
1114                 <xs:annotation>
1115                     <xs:documentation>
1116                         30% dos alunos normalmente atingem este valor.
1117                         BOM - na geralidade um trabalho com qualidade, mas com erros importantes.
1118                     </xs:documentation>
1119                 </xs:annotation>
1120             </xs:enumeration>
1121             <xs:enumeration value="D">
1122                 <xs:annotation>
1123                     <xs:documentation>
1124                         25% dos alunos normalmente atingem este valor.
1125                         SATIFATÓRIO - interessante, mas com limitações significativas.
1126                     </xs:documentation>
1127                 </xs:annotation>
1128             </xs:enumeration>
1129             <xs:enumeration value="E">
1130                 <xs:annotation>
1131                     <xs:documentation>
1132                         10% dos alunos normalmente atingem este valor.
1133                         SUFICIENTE - o desempenho cumpre os critérios mínimos.
1134                     </xs:documentation>
1135                 </xs:annotation>
1136             </xs:enumeration>
1137             <xs:enumeration value="FX">
1138                 <xs:annotation>
1139                     <xs:documentation>
1140                         INSUFICIENTE - algum trabalho extra é necessário para obter os créditos.
1141                     </xs:documentation>
1142                 </xs:annotation>
1143             </xs:enumeration>
1144             <xs:enumeration value="F">
1145                 <xs:annotation>
1146                     <xs:documentation>
1147                         INSUFICIENTE - considerável trabalho extra é necessário para obter os créditos.
1148                     </xs:documentation>
1149                 </xs:annotation>
1150             </xs:enumeration>

```

```

1151         </xs:restriction>
1152     </xs:simpleType>
1153
1154     <!-- EquivalenceInformationType -->
1155     <xs:complexType name="EquivalenceInformationType">
1156         <xs:annotation>
1157             <xs:documentation>
1158                 Informação sobre uma equivalência.
1159             </xs:documentation>
1160         </xs:annotation>
1161         <xs:sequence>
1162             <xs:element name="Degree" type="DegreeType" />
1163             <xs:element name="Course" type="CourseType" />
1164             <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
1165         </xs:sequence>
1166     </xs:complexType>
1167
1168     <!-- ##### -->
1169     <!-- ThesisType -->
1170     <!-- ##### -->
1171     <xs:complexType name="ThesisType">
1172         <xs:sequence>
1173             <xs:element name="ThesisDissertationTitle" type="xs:string" />
1174             <xs:element name="ThesisDissertationSupervisor" type="xs:string" minOccurs="0" maxOccurs="unbounded" />
1175             <xs:element name="ThesisDissertationEvaluation" type="ThesisDissertationEvaluationType" minOccurs="0" />
1176             <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
1177         </xs:sequence>
1178     </xs:complexType>
1179
1180     <!-- ThesisDissertationGradeType -->
1181     <xs:simpleType name="ThesisDissertationEvaluationType">
1182         <xs:restriction base="xs:string" />
1183     </xs:simpleType>
1184
1185     <!-- ##### -->
1186     <!-- HealthRecordType -->
1187     <!-- ##### -->
1188     <xs:complexType name="HealthRecordType">
1189         <xs:annotation>
1190             <xs:documentation>
1191                 Informação sobre um registo de saúde.
1192             </xs:documentation>
1193         </xs:annotation>
1194         <xs:sequence>
1195             <xs:element name="Examination" type="ExaminationType" minOccurs="0" maxOccurs="unbounded" />
1196             <xs:element name="Immunization" type="ImmunizationType" minOccurs="0" maxOccurs="unbounded" />
1197             <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
1198         </xs:sequence>
1199     </xs:complexType>
1200
1201     <!-- ExaminationType -->
1202     <xs:complexType name="ExaminationType">
1203         <xs:annotation>
1204             <xs:documentation>
1205                 Informação sobre um exame médico.
1206             </xs:documentation>
1207         </xs:annotation>
1208         <xs:sequence>
1209             <xs:element name="ExaminationName" type="ExaminationNameType" />
1210             <xs:element name="ExaminationCode" type="ExaminationCodeType" minOccurs="0" />
1211             <xs:element name="ExaminationDate" type="ExaminationDateType" minOccurs="0" />
1212             <xs:element name="ExaminationResult" type="ExaminationResultType" minOccurs="0" />
1213             <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
1214         </xs:sequence>
1215     </xs:complexType>
1216
1217     <!-- ExaminationNameType -->
1218     <xs:simpleType name="ExaminationNameType">
1219         <xs:restriction base="xs:string" />
1220     </xs:simpleType>
1221
1222     <!-- ExaminationCodeType -->
1223     <xs:simpleType name="ExaminationCodeType">
1224         <xs:annotation>
1225             <xs:documentation>
1226                 Código identificativo do exame, no sistema adoptado pela instituição.
1227             </xs:documentation>
1228         </xs:annotation>
1229         <xs:restriction base="xs:string" />
1230     </xs:simpleType>
1231
1232     <!-- ExaminationDateType -->
1233     <xs:simpleType name="ExaminationDateType">
1234         <xs:restriction base="xs:date" />
1235     </xs:simpleType>
1236
1237     <!-- ExaminationResultType -->
1238     <xs:simpleType name="ExaminationResultType">
1239         <xs:restriction base="xs:string" />
1240     </xs:simpleType>
1241
1242     <!-- ImmunizationType -->

```



```

1243 <xs:complexType name="ImmunizationType">
1244 <xs:annotation>
1245 <xs:documentation>
1246 Informação sobre uma vacina.
1247 </xs:documentation>
1248 </xs:annotation>
1249 <xs:sequence>
1250 <xs:element name="ImmunizationName" type="ImmunizationNameType"/>
1251 <xs:element name="ImmunizationCode" type="ImmunizationCodeType" minOccurs="0"/>
1252 <xs:element name="ImmunizationDate" type="ImmunizationDateType" minOccurs="0"/>
1253 <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
1254 </xs:sequence>
1255 </xs:complexType>
1256
1257 <!-- ImmunizationNameType -->
1258 <xs:simpleType name="ImmunizationNameType">
1259 <xs:restriction base="xs:string"/>
1260 </xs:simpleType>
1261
1262 <!-- ImmunizationCodeType -->
1263 <xs:simpleType name="ImmunizationCodeType">
1264 <xs:annotation>
1265 <xs:documentation>
1266 Código identificativo da vacina, no sistema adoptado pela instituição.
1267 </xs:documentation>
1268 </xs:annotation>
1269 <xs:restriction base="xs:string"/>
1270 </xs:simpleType>
1271
1272 <!-- ImmunizationDateType -->
1273 <xs:simpleType name="ImmunizationDateType">
1274 <xs:restriction base="xs:date"/>
1275 </xs:simpleType>
1276
1277 <!-- ##### -->
1278 <!-- MilitaryRecordType -->
1279 <!-- ##### -->
1280 <xs:complexType name="MilitaryRecordType">
1281 <xs:annotation>
1282 <xs:documentation>
1283 Informação sobre a situação militar.
1284 </xs:documentation>
1285 </xs:annotation>
1286 <xs:sequence>
1287 <xs:element name="MilitaryStatus" type="MilitaryStatusType"/>
1288 <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
1289 </xs:sequence>
1290 </xs:complexType>
1291
1292 <!-- MilitaryStatusType -->
1293 <xs:simpleType name="MilitaryStatusType">
1294 <xs:restriction base="xs:string">
1295 <xs:enumeration value="PP">
1296 <xs:annotation>
1297 <xs:documentation>Adiado / Postponed.</xs:documentation>
1298 </xs:annotation>
1299 </xs:enumeration>
1300 <xs:enumeration value="DN">
1301 <xs:annotation>
1302 <xs:documentation>Cumprida / Accomplished (Done).</xs:documentation>
1303 </xs:annotation>
1304 </xs:enumeration>
1305 <xs:enumeration value="RS">
1306 <xs:annotation>
1307 <xs:documentation>Reserva / Reserve.</xs:documentation>
1308 </xs:annotation>
1309 </xs:enumeration>
1310 <xs:enumeration value="DA">
1311 <xs:annotation>
1312 <xs:documentation>Inapto / Disable.</xs:documentation>
1313 </xs:annotation>
1314 </xs:enumeration>
1315 <xs:enumeration value="NA">
1316 <xs:annotation>
1317 <xs:documentation>Não Aplicável / Not Applicable.</xs:documentation>
1318 </xs:annotation>
1319 </xs:enumeration>
1320 <xs:enumeration value="UK">
1321 <xs:annotation>
1322 <xs:documentation>Desconhecido / Unknown.</xs:documentation>
1323 </xs:annotation>
1324 </xs:enumeration>
1325 </xs:restriction>
1326 </xs:simpleType>
1327
1328 <!-- ##### -->
1329 <!-- AcademicAwardType -->
1330 <!-- ##### -->
1331 <xs:complexType name="AcademicAwardType">
1332 <xs:annotation>
1333 <xs:documentation>
1334 Prémio académico. Recebido no contexto de uma matrícula numa instituição.
1335 </xs:documentation>
1336 </xs:annotation>

```

```

1337         <xs:sequence>
1338             <xs:element name="AcademicAwardName" type="AcademicAwardNameType" minOccurs="0" />
1339             <xs:element name="AcademicAwardDate" type="AcademicAwardDateType" minOccurs="0" />
1340             <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
1341         </xs:sequence>
1342     </xs:complexType>
1343
1344     <!-- AcademicAwardType -->
1345     <xs:simpleType name="AcademicAwardNameType">
1346         <xs:restriction base="xs:string" />
1347     </xs:simpleType>
1348
1349     <!-- AcademicAwardDateType -->
1350     <xs:simpleType name="AcademicAwardDateType">
1351         <xs:restriction base="xs:date" />
1352     </xs:simpleType>
1353
1354     <!-- ##### -->
1355     <!-- ExtraActivitiesType -->
1356     <!-- ##### -->
1357     <xs:complexType name="ExtraActivityType">
1358         <xs:annotation>
1359             <xs:documentation>
1360                 Tipo para representar uma actividade extra de qualquer índole.
1361             </xs:documentation>
1362         </xs:annotation>
1363         <xs:sequence>
1364             <xs:element name="ExtraActivityName" type="ExtraActivityNameType" />
1365             <xs:element name="ExtraActivityStartDate" type="xs:date" minOccurs="0" />
1366             <xs:element name="ExtraActivityEndDate" type="xs:date" minOccurs="0" />
1367             <xs:element name="ExtraActivityDescription" type="ExtraActivityDescriptionType" minOccurs="0"
1368                 maxOccurs="unbounded" />
1369             <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded" />
1370         </xs:sequence>
1371     </xs:complexType>
1372
1373     <!-- ExtraActivityNameType -->
1374     <xs:simpleType name="ExtraActivityNameType">
1375         <xs:restriction base="xs:string" />
1376     </xs:simpleType>
1377
1378     <!-- ExtraActivityDescriptionType -->
1379     <xs:simpleType name="ExtraActivityDescriptionType">
1380         <xs:restriction base="xs:string" />
1381     </xs:simpleType>
1382 </xs:schema>

```

Apêndice C

Agregação de Estatísticas sobre Cursos

C.1 Especificação do Elemento DegreeStatistics

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <xs:schema targetNamespace="http://schemas.up.pt/DegreeStatistics" xmlns="http://schemas.up.pt/DegreeStatistics"
   xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" attributeFormDefault="unqualified">
3
4     <!-- ##### -->
5     <!-- DegreeStatistics -->
6     <!-- ##### -->
7     <xs:element name="DegreeStatistics" type="DegreeStatisticsType"/>
8
9     <!-- ##### -->
10    <!-- DegreeStatisticsType -->
11    <!-- ##### -->
12    <xs:complexType name="DegreeStatisticsType">
13        <xs:annotation>
14            <xs:documentation>
15                Estatísticas sobre um curso académico, relativas a um edição bem identificada.
16            </xs:documentation>
17        </xs:annotation>
18        <xs:sequence>
19            <xs:element name="DocumentInfo" type="DocumentInfoType"/>
20            <xs:element name="DegreeIdentification" type="DegreeIdentificationType"/>
21            <xs:element name="DegreeAcronym" type="DegreeAcronymType" minOccurs="0">
22                <xs:annotation>
23                    <xs:documentation>
24                        Sigla do curso.
25                    </xs:documentation>
26                </xs:annotation>
27            </xs:element>
28            <xs:element name="DegreeName" type="DegreeNameType">
29                <xs:annotation>
30                    <xs:documentation>
31                        Nome do curso.
32                    </xs:documentation>
33                </xs:annotation>
34            </xs:element>
35            <xs:element name="DegreeStatisticsStartDate" type="xs:date">
36                <xs:annotation>
37                    <xs:documentation>
38                        Data de início do período em relação ao qual são apresentadas as
39                        informações.
40                    </xs:documentation>
41                </xs:annotation>
42            </xs:element>
43            <xs:element name="DegreeStatisticsEndDate" type="xs:date">
44                <xs:annotation>
45                    <xs:documentation>
46                        Data de fim do período em relação ao qual são apresentadas as
47                        informações.
48                    </xs:documentation>
49                </xs:annotation>
50            </xs:element>
51            <xs:element name="DegreeFaculty" type="PersonsNumberType" minOccurs="0">
52                <xs:annotation>
53                    <xs:documentation>
54                        Docentes do curso.
55                    </xs:documentation>
56                </xs:annotation>
57            </xs:element>
58            <xs:element name="AdmissionVacancies" type="xs:integer" minOccurs="0">
59                <xs:annotation>
60                    <xs:documentation>
61                        Numeros Clausus do curso.
62                    </xs:documentation>
63                </xs:annotation>
64            </xs:element>
65        </xs:sequence>
66    </xs:complexType>
67 </xs:schema>
```

```

60         </xs:documentation>
61     </xs:annotation>
62 </xs:element>
63 <xs:element name="AcceptedApplicantsNumber" type="xs:integer" minOccurs="0">
64     <xs:annotation>
65         <xs:documentation>
66             Número de candidatos ao curso aceites.
67         </xs:documentation>
68     </xs:annotation>
69 </xs:element>
70 <xs:element name="AcceptedApplicantsGradeAverage" type="GradeValueType" minOccurs="0">
71     <xs:annotation>
72         <xs:documentation>
73             Média aritmética das notas de candidatura dos candidatos aceites.
74             Representada na escala da instituição.
75         </xs:documentation>
76     </xs:annotation>
77 </xs:element>
78 <xs:element name="AcceptedApplicantsGradeMedian" type="GradeValueType" minOccurs="0">
79     <xs:annotation>
80         <xs:documentation>
81             Mediana das notas de candidatura dos candidatos aceites.
82             Representada na escala da instituição.
83         </xs:documentation>
84     </xs:annotation>
85 </xs:element>
86 <xs:element name="AcceptedApplicantsGradeStandardDeviation" type="xs:decimal" minOccurs="0">
87     <xs:annotation>
88         <xs:documentation>
89             Desvio padrão das notas de candidatura dos candidatos aceites.
90         </xs:documentation>
91     </xs:annotation>
92 </xs:element>
93 <xs:element name="AcceptedApplicantsGradeMinimum" type="GradeValueType" minOccurs="0">
94     <xs:annotation>
95         <xs:documentation>
96             Mínimo das notas de candidatura dos candidatos aceites.
97             Representada na escala da instituição.
98         </xs:documentation>
99     </xs:annotation>
100 </xs:element>
101 <xs:element name="AcceptedApplicantsGradeMaximum" type="GradeValueType" minOccurs="0">
102     <xs:annotation>
103         <xs:documentation>
104             Máximo das notas de candidatura dos candidatos aceites.
105             Representada na escala da instituição.
106         </xs:documentation>
107     </xs:annotation>
108 </xs:element>
109 <xs:element name="EnrolledStudents" type="PersonsNumberType" minOccurs="0">
110     <xs:annotation>
111         <xs:documentation>
112             Alunos inscritos no curso.
113         </xs:documentation>
114     </xs:annotation>
115 </xs:element>
116 <xs:element name="EnrolledStudentsByCountry" type="PersonsByCountryType" minOccurs="0">
117     <xs:annotation>
118         <xs:documentation>
119             Alunos inscritos por países.
120         </xs:documentation>
121     </xs:annotation>
122 </xs:element>
123 <xs:element name="EnrolledStudentsByAge" type="PersonsByAgeType" minOccurs="0">
124     <xs:annotation>
125         <xs:documentation>
126             Alunos inscritos por idades.
127         </xs:documentation>
128     </xs:annotation>
129 </xs:element>
130 <xs:element name="EnrolledStudentsByCurricularPeriod" type="PersonsByCurricularPeriodType"
131     minOccurs="0">
132     <xs:annotation>
133         <xs:documentation>
134             Alunos inscritos por ano curricular do curso.
135         </xs:documentation>
136     </xs:annotation>
137 </xs:element>
138 <xs:element name="EnrolledStudentsByEnrollmentsByCurricularPeriod" type="
139     PersonsByEnrollmentsByCurricularPeriodType" minOccurs="0">
140     <xs:annotation>
141         <xs:documentation>
142             Número de inscrições por período curricular.
143             Exemplo: No 1o ano há X alunos inscritos pela 1a vez, Y alunos
144             inscritos pela 2a vez, ...
145         </xs:documentation>
146     </xs:annotation>
147 </xs:element>
148 <xs:element name="Graduates" type="PersonsNumberType" minOccurs="0">
149     <xs:annotation>
150         <xs:documentation>
151             Número de diplomados.
152         </xs:documentation>
153     </xs:annotation>
154 </xs:element>
155 </xs:annotation>

```

```

151         </xs:element>
152         <xs:element name="GraduatesByCountry" type="PersonsByCountryType" minOccurs="0">
153             <xs:annotation>
154                 <xs:documentation>
155                     Número de diplomados por país.
156                 </xs:documentation>
157             </xs:annotation>
158         </xs:element>
159         <xs:element name="GraduatesByAge" type="PersonsByAgeType" minOccurs="0">
160             <xs:annotation>
161                 <xs:documentation>
162                     Número de diplomados por idade.
163                 </xs:documentation>
164             </xs:annotation>
165         </xs:element>
166         <xs:element name="GraduatesByFinalGradeValue" type="PersonsByFinalGradeValueType" minOccurs="0">
167             <xs:annotation>
168                 <xs:documentation>
169                     Número de diplomados por média final.
170                 </xs:documentation>
171             </xs:annotation>
172         </xs:element>
173         <xs:element name="GraduatesByTotalEnrollments" type="PersonsByTotalEnrollmentsType" minOccurs="0">
174             <xs:annotation>
175                 <xs:documentation>
176                     Número de diplomados por número total de inscrições.
177                 </xs:documentation>
178             </xs:annotation>
179         </xs:element>
180         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
181     </xs:sequence>
182 </xs:complexType>
183
184 <!-- ##### -->
185 <!-- DocumentInfoType -->
186 <!-- ##### -->
187 <xs:complexType name="DocumentInfoType">
188     <xs:sequence>
189         <xs:element name="DocumentCode"/>
190         <xs:element name="DocumentIssueDate"/>
191         <xs:element name="DocumentIssuer"/>
192         <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
193     </xs:sequence>
194 </xs:complexType>
195
196 <!-- ##### -->
197 <!-- DegreeIdentificationType -->
198 <!-- ##### -->
199 <xs:complexType name="DegreeIdentificationType">
200     <xs:annotation>
201         <xs:documentation>
202             Identificador completo do curso.
203             Construção: País + Instituição + Curso.
204         </xs:documentation>
205     </xs:annotation>
206     <xs:sequence>
207         <xs:element name="CountryCode" type="CountryCodeType"/>
208         <xs:element name="InstitutionCode" type="InstitutionCodeType"/>
209         <xs:element name="DegreeCode" type="DegreeCodeType"/>
210     </xs:sequence>
211 </xs:complexType>
212
213 <!-- CountryCodeType -->
214 <xs:simpleType name="CountryCodeType">
215     <xs:annotation>
216         <xs:documentation>
217             Identificador de um país usando o código numérico da norma ISO 3166-1. Exemplo:
218             Portugal = 620.
219         </xs:documentation>
220     </xs:annotation>
221     <xs:restriction base="xs:string">
222         <xs:pattern value="\d\d\d"/>
223     </xs:restriction>
224 </xs:simpleType>
225
226 <!-- InstitutionCodeType -->
227 <xs:simpleType name="InstitutionCodeType">
228     <xs:annotation>
229         <xs:documentation>
230             Código da instituição. O sistema usado deve ser definido para cada país.
231         </xs:documentation>
232     </xs:annotation>
233     <xs:restriction base="xs:string"/>
234 </xs:simpleType>
235
236 <!-- DegreeCodeType -->
237 <xs:simpleType name="DegreeCodeType">
238     <xs:annotation>
239         <xs:documentation>
240             Identificador do curso. O sistema de identificação usado é definido pela instituição.
241         </xs:documentation>
242     </xs:annotation>

```

```

242     <xs:restriction base="xs:string" />
243 </xs:simpleType>
244
245 <!-- DegreeAcronymType -->
246 <xs:simpleType name="DegreeAcronymType">
247     <xs:restriction base="xs:string" />
248 </xs:simpleType>
249
250 <!-- DegreeNameType -->
251 <xs:simpleType name="DegreeNameType">
252     <xs:restriction base="xs:string" />
253 </xs:simpleType>
254
255 <!-- ##### -->
256 <!-- PersonsByCountryType -->
257 <!-- ##### -->
258 <xs:complexType name="PersonsByCountryType">
259     <xs:sequence>
260         <xs:element name="CountryFrequency" maxOccurs="unbounded">
261             <xs:complexType>
262                 <xs:sequence>
263                     <xs:element name="CountryCode" type="CountryCodeType" />
264                     <xs:element name="PersonsNumber" type="PersonsNumberType" />
265                 </xs:sequence>
266             </xs:complexType>
267         </xs:element>
268     </xs:sequence>
269 </xs:complexType>
270
271 <!-- ##### -->
272 <!-- PersonsByAgeType -->
273 <!-- ##### -->
274 <xs:complexType name="PersonsByAgeType">
275     <xs:sequence>
276         <xs:element name="AgeFrequency" maxOccurs="unbounded">
277             <xs:complexType>
278                 <xs:sequence>
279                     <xs:element name="Age" type="xs:integer" />
280                     <xs:element name="PersonsNumber" type="PersonsNumberType" />
281                 </xs:sequence>
282             </xs:complexType>
283         </xs:element>
284     </xs:sequence>
285 </xs:complexType>
286
287 <!-- ##### -->
288 <!-- PersonsByCurricularYearType -->
289 <!-- ##### -->
290 <xs:complexType name="PersonsByCurricularPeriodType">
291     <xs:sequence>
292         <xs:element name="CurricularPeriodFrequency" maxOccurs="unbounded">
293             <xs:complexType>
294                 <xs:sequence>
295                     <xs:element name="CurricularPeriod" type="CurricularPeriodType">
296                         <xs:annotation>
297                             <xs:documentation>
298                                 Período curricular.
299                                 Exemplo: 1o ano, 2o ano; 1o semestre, 2o
300                                 semestre;
301                                 Representa a *ordem* deste período, no
302                                 conjunto total de períodos.
303                             </xs:documentation>
304                         </xs:annotation>
305                     </xs:element>
306                     <xs:element name="CurricularPeriodDuration" type="
307                         CurricularPeriodDurationType">
308                         <xs:annotation>
309                             <xs:documentation>
310                                 Duração do período acadêmico.
311                             </xs:documentation>
312                         </xs:annotation>
313                     </xs:element>
314                     <xs:element name="PersonsNumber" type="PersonsNumberType" />
315                 </xs:sequence>
316             </xs:complexType>
317         </xs:element>
318     </xs:sequence>
319 </xs:complexType>
320
321 <!-- ##### -->
322 <!-- PersonsByEnrollmentsByCurricularPeriodType -->
323 <!-- ##### -->
324 <xs:complexType name="PersonsByEnrollmentsByCurricularPeriodType">
325     <xs:sequence>
326         <xs:element name="CurricularPeriodFrequency" maxOccurs="unbounded">
327             <xs:complexType>
328                 <xs:sequence>
329                     <xs:element name="CurricularPeriod" type="CurricularPeriodType" />
330                     <xs:element name="CurricularPeriodDuration" type="
331                         CurricularPeriodDurationType" />
332                     <xs:element name="EnrollmentsFrequency" maxOccurs="unbounded">
333                         <xs:complexType>
334                             <xs:sequence>

```

```

331         <xs:element name="Enrollments" type="
332             xs:integer"/>
333         <xs:element name="PersonsNumber" type="
334             PersonsNumberType"/>
335     </xs:sequence>
336 </xs:complexType>
337 </xs:element>
338     </xs:sequence>
339 </xs:complexType>
340 </xs:complexType>
341
342 <!-- ##### -->
343 <!-- PersonsByFinalGradeValueType -->
344 <!-- ##### -->
345 <xs:complexType name="PersonsByFinalGradeValueType">
346     <xs:sequence>
347         <xs:element name="FinalGradeValueFrequency" maxOccurs="unbounded">
348             <xs:complexType>
349                 <xs:sequence>
350                     <xs:element name="FinalGradeValue" type="GradeValueType"/>
351                     <xs:element name="PersonsNumber" type="PersonsNumberType"/>
352                 </xs:sequence>
353             </xs:complexType>
354         </xs:element>
355     </xs:sequence>
356 </xs:complexType>
357
358 <!-- ##### -->
359 <!-- PersonsByTotalEnrollmentsType -->
360 <!-- ##### -->
361 <xs:complexType name="PersonsByTotalEnrollmentsType">
362     <xs:sequence>
363         <xs:element name="TotalEnrollmentsFrequency" maxOccurs="unbounded">
364             <xs:complexType>
365                 <xs:sequence>
366                     <xs:element name="TotalEnrollments" type="xs:integer"/>
367                     <xs:element name="PersonsNumber" type="PersonsNumberType"/>
368                 </xs:sequence>
369             </xs:complexType>
370         </xs:element>
371     </xs:sequence>
372 </xs:complexType>
373
374 <!-- ##### -->
375 <!-- CurricularPeriodType -->
376 <!-- ##### -->
377 <xs:simpleType name="CurricularPeriodType">
378     <xs:annotation>
379         <xs:documentation>
380             Período curricular em questão: primeiro, segundo, terceiro, etc.
381         </xs:documentation>
382     </xs:annotation>
383     <xs:restriction base="xs:integer"/>
384 </xs:simpleType>
385
386 <!-- ##### -->
387 <!-- CurricularPeriodDurationType -->
388 <!-- ##### -->
389 <xs:simpleType name="CurricularPeriodDurationType">
390     <xs:annotation>
391         <xs:documentation>
392             Duração do período curricular.
393         </xs:documentation>
394     </xs:annotation>
395     <xs:restriction base="xs:duration"/>
396 </xs:simpleType>
397
398 <!-- ##### -->
399 <!-- PersonsNumberType -->
400 <!-- ##### -->
401 <xs:complexType name="PersonsNumberType">
402     <xs:annotation>
403         <xs:documentation>
404             Número de pessoas por sexo.
405         </xs:documentation>
406     </xs:annotation>
407     <xs:sequence>
408         <xs:element name="Total" type="xs:integer" minOccurs="0">
409             <xs:annotation>
410                 <xs:documentation>
411                     O TotalStudents permite ter um valor, mesmo sem os dados parcelares
412                     segundo os sexos.
413                 </xs:documentation>
414             </xs:annotation>
415         <xs:element name="Female" type="xs:integer" minOccurs="0"/>
416         <xs:element name="Male" type="xs:integer" minOccurs="0"/>
417     </xs:sequence>
418 </xs:complexType>
419
420 <!-- ##### -->
421 <!-- GradeValueType -->

```

```
422      <!-- ##### -->
423      <xs:simpleType name="GradeValueType">
424        <xs:restriction base="xs:string"/>
425      </xs:simpleType>
426
427      <!-- ##### -->
428      <!-- NoteType -->
429      <!-- ##### -->
430      <xs:simpleType name="NoteType">
431        <xs:restriction base="xs:string"/>
432      </xs:simpleType>
433 </xs:schema>
```


Apêndice D

Concentração de Informação sobre Recursos Humanos

D.1 Especificação do Elemento StaffRecord

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <xs:schema targetNamespace="http://schemas.up.pt/StaffRecord" xmlns="http://schemas.up.pt/StaffRecord" xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" attributeFormDefault="unqualified">
3
4     <!-- ##### -->
5     <!-- StaffRecord -->
6     <!-- ##### -->
7     <xs:element name="StaffRecord" type="StaffRecordType"/>
8
9     <!-- ##### -->
10    <!-- StaffRecordType -->
11    <!-- ##### -->
12    <xs:complexType name="StaffRecordType">
13        <xs:annotation>
14            <xs:documentation>
15                Funcionário.
16            </xs:documentation>
17        </xs:annotation>
18        <xs:sequence>
19            <xs:element name="StaffIdentification" type="StaffIdentificationType"/>
20            <xs:element name="StaffAcronym" type="StaffAcronymType" minOccurs="0"/>
21            <xs:element name="Person" type="PersonType"/>
22            <xs:element name="StaffCategory" type="StaffCategoryType" minOccurs="0"/>
23            <xs:element name="StaffStatus" type="StaffStatusType" minOccurs="0"/>
24            <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
25        </xs:sequence>
26    </xs:complexType>
27
28    <!-- ##### -->
29    <!-- StaffIdentificationType -->
30    <!-- ##### -->
31    <xs:complexType name="StaffIdentificationType">
32        <xs:annotation>
33            <xs:documentation>
34                Identificação de um funcionário.
35                A identificação do funcionário é composta usando o código do país, o código da
36                instituição e o código do funcionário.
37            </xs:documentation>
38        </xs:annotation>
39        <xs:sequence>
40            <xs:element name="CountryCode" type="CountryCodeType"/>
41            <xs:element name="InstitutionCode" type="InstitutionCodeType"/>
42            <xs:element name="StaffCode" type="StaffCodeType"/>
43        </xs:sequence>
44    </xs:complexType>
45
46    <!-- CountryCodeType -->
47    <xs:simpleType name="CountryCodeType">
48        <xs:annotation>
49            <xs:documentation>
50                Identificador de um país usando o código numérico da norma ISO 3166-1. Exemplo:
51                Portugal = 620.
52            </xs:documentation>
53        </xs:annotation>
54        <xs:restriction base="xs:string">
55            <xs:pattern value="\d\d\d"/>
56        </xs:restriction>
57    </xs:simpleType>
```

```

56 <!-- InstitutionCodeType -->
57 <xs:simpleType name="InstitutionCodeType">
58   <xs:annotation>
59     <xs:documentation>
60       Código da instituição no contexto do país. O sistema usado deve ser definido para cada
61       país.
62     </xs:documentation>
63   </xs:annotation>
64   <xs:restriction base="xs:string"/>
65 </xs:simpleType>
66
67 <!-- StaffCodeType -->
68 <xs:simpleType name="StaffCodeType">
69   <xs:annotation>
70     <xs:documentation>
71       Identificador de cada funcionário na instituição.
72     </xs:documentation>
73   </xs:annotation>
74   <xs:restriction base="xs:string"/>
75 </xs:simpleType>
76
77 <!-- ##### -->
78 <!-- StaffAcronymType -->
79 <!-- ##### -->
80 <xs:simpleType name="StaffAcronymType">
81   <xs:restriction base="xs:string"/>
82 </xs:simpleType>
83
84 <!-- ##### -->
85 <!-- PersonType -->
86 <!-- ##### -->
87 <xs:complexType name="PersonType">
88   <xs:annotation>
89     <xs:documentation>
90       Definição do tipo para representar uma pessoa.
91     </xs:documentation>
92   </xs:annotation>
93   <xs:sequence>
94     <xs:element name="IdentificationCode" type="IdentificationCodeType" minOccurs="0" maxOccurs="
95       unbounded">
96       <xs:annotation>
97         <xs:documentation>
98           Uma pessoa pode ter vários códigos de identificação. Por exemplo, BI,
99           NIF, etc.
100         </xs:documentation>
101       </xs:annotation>
102     </xs:element>
103     <xs:element name="Name" type="NameType"/>
104     <xs:element name="Birth" type="BirthType" minOccurs="0"/>
105     <xs:element name="Deceased" type="DeceasedType" minOccurs="0">
106       <xs:annotation>
107         <xs:documentation>
108           Este elemento existe se a pessoa já faleceu.
109         </xs:documentation>
110       </xs:annotation>
111     </xs:element>
112     <xs:element name="Gender" type="GenderType" minOccurs="0"/>
113     <xs:element name="MaritalStatus" type="MaritalStatusType" minOccurs="0"/>
114     <xs:element name="Citizenship" type="CitizenshipType" minOccurs="0"/>
115     <xs:element name="Residency" type="AddressType" minOccurs="0">
116       <xs:annotation>
117         <xs:documentation>
118           Residência "oficial", por isso só pode ter uma instância.
119         </xs:documentation>
120       </xs:annotation>
121     </xs:element>
122     <xs:element name="Contact" type="ContactType" minOccurs="0" maxOccurs="unbounded">
123       <xs:annotation>
124         <xs:documentation>
125           Uma pessoa pode ter vários contactos de vários tipos. Por exemplo:
126           telemóvel, telefone de casa, telefone do emprego, etc.
127         </xs:documentation>
128       </xs:annotation>
129     </xs:element>
130     <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
131   </xs:sequence>
132 </xs:complexType>
133
134 <!-- NameType -->
135 <xs:complexType name="NameType">
136   <xs:annotation>
137     <xs:documentation>
138       Nome de uma pessoa.
139     </xs:documentation>
140   </xs:annotation>
141   <xs:sequence>
142     <xs:element name="Title" type="NameTitleType" minOccurs="0"/>
143     <xs:element name="FirstName" type="FirstNameType" minOccurs="0"/>
144     <xs:element name="MiddleName" type="MiddleNameType" minOccurs="0" maxOccurs="unbounded"/>
145     <xs:element name="LastName" type="LastNameType" minOccurs="0"/>
146     <xs:element name="FullName" type="FullNameType"/>
147     <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
148   </xs:sequence>

```

```

146     </xs:complexType>
147
148     <!-- NameTitleType -->
149     <xs:simpleType name="NameTitleType">
150         <xs:restriction base="xs:string"/>
151     </xs:simpleType>
152
153     <!-- FirstNameType -->
154     <xs:simpleType name="FirstNameType">
155         <xs:restriction base="xs:string"/>
156     </xs:simpleType>
157
158     <!-- MiddleNameType -->
159     <xs:simpleType name="MiddleNameType">
160         <xs:restriction base="xs:string"/>
161     </xs:simpleType>
162
163     <!-- LastNameType -->
164     <xs:simpleType name="LastNameType">
165         <xs:restriction base="xs:string"/>
166     </xs:simpleType>
167
168     <!-- FullNameType -->
169     <xs:simpleType name="FullNameType">
170         <xs:restriction base="xs:string"/>
171     </xs:simpleType>
172
173     <!-- BirthType -->
174     <xs:complexType name="BirthType">
175         <xs:sequence>
176             <xs:element name="BirthDate" type="xs:date"/>
177             <xs:element name="BirthCity" type="CityType" minOccurs="0"/>
178             <xs:element name="BirthCountry" type="CountryType" minOccurs="0"/>
179             <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
180         </xs:sequence>
181     </xs:complexType>
182
183     <!-- DeceasedType -->
184     <xs:complexType name="DeceasedType">
185         <xs:sequence>
186             <xs:element name="DeceasedIndicator" type="xs:boolean"/>
187             <xs:element name="DeceasedDate" type="xs:date" minOccurs="0"/>
188             <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
189         </xs:sequence>
190     </xs:complexType>
191
192     <!-- GenderType -->
193     <xs:simpleType name="GenderType">
194         <xs:restriction base="xs:string">
195             <xs:enumeration value="F">
196                 <xs:annotation>
197                     <xs:documentation>Female / Feminino.</xs:documentation>
198                 </xs:annotation>
199             </xs:enumeration>
200             <xs:enumeration value="M">
201                 <xs:annotation>
202                     <xs:documentation>Male / Masculino.</xs:documentation>
203                 </xs:annotation>
204             </xs:enumeration>
205             <xs:enumeration value="U">
206                 <xs:annotation>
207                     <xs:documentation>Unspecified / Não especificado.</xs:documentation>
208                 </xs:annotation>
209             </xs:enumeration>
210         </xs:restriction>
211     </xs:simpleType>
212
213     <!-- MaritalStatusType -->
214     <xs:simpleType name="MaritalStatusType">
215         <xs:restriction base="xs:string">
216             <xs:enumeration value="S">
217                 <xs:annotation>
218                     <xs:documentation>Single / Solteiro.</xs:documentation>
219                 </xs:annotation>
220             </xs:enumeration>
221             <xs:enumeration value="M">
222                 <xs:annotation>
223                     <xs:documentation>Married / Casado.</xs:documentation>
224                 </xs:annotation>
225             </xs:enumeration>
226             <xs:enumeration value="D">
227                 <xs:annotation>
228                     <xs:documentation>Divorced / Divorciado.</xs:documentation>
229                 </xs:annotation>
230             </xs:enumeration>
231             <xs:enumeration value="W">
232                 <xs:annotation>
233                     <xs:documentation>Widowed / Viúvo.</xs:documentation>
234                 </xs:annotation>
235             </xs:enumeration>
236         </xs:restriction>
237     </xs:simpleType>
238
239     <!-- CitizenshipType -->

```

```

240 <xs:simpleType name="CitizenshipType">
241   <xs:restriction base="CountryType"/>
242 </xs:simpleType>
243
244 <!-- IdentificationCodeType -->
245 <xs:complexType name="IdentificationCodeType">
246   <xs:annotation>
247     <xs:documentation>
248       Definição do tipo para representar números de identificação de cartões. Por exemplo:
249       bilhete de identidade, cartão de contribuinte, etc.
250     </xs:documentation>
251   </xs:annotation>
252   <xs:sequence>
253     <xs:element name="IdentificationCodeCode" type="IdentificationCodeCodeType" minOccurs="0">
254       <xs:annotation>
255         <xs:documentation>
256           Código identificativo do sistema de identificação usado.
257           Usar nos casos em que o sistema de identificação é um dos tipos pré-
258             definidos.
259         </xs:documentation>
260       </xs:annotation>
261     </xs:element>
262     <xs:element name="IdentificationCodeDescription" type="IdentificationCodeDescriptionType"/>
263     <xs:element name="IdentificationCodeValue" type="IdentificationCodeValueType"/>
264     <xs:element name="IdentificationCodeIssuer" type="IdentificationCodeIssuerType"/>
265     <xs:element name="IdentificationCodeIssueDate" type="IdentificationCodeIssueDateType"
266       minOccurs="0"/>
267     <xs:element name="IdentificationCodeExpirationDate" type="IdentificationCodeExpirationDateType"
268       minOccurs="0"/>
269     <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
270   </xs:sequence>
271 </xs:complexType>
272
273 <!-- IdentificationCodeCodeType -->
274 <xs:simpleType name="IdentificationCodeCodeType">
275   <xs:annotation>
276     <xs:documentation>
277       Código identificativo do sistema de identificação usado.
278     </xs:documentation>
279   </xs:annotation>
280   <xs:restriction base="xs:string">
281     <xs:enumeration value="BI">
282       <xs:annotation>
283         <xs:documentation>Bilhete de Identidade.</xs:documentation>
284       </xs:annotation>
285     </xs:enumeration>
286     <xs:enumeration value="NIF">
287       <xs:annotation>
288         <xs:documentation>Número de Identificação Fiscal.</xs:documentation>
289       </xs:annotation>
290     </xs:enumeration>
291     <xs:enumeration value="NCC">
292       <xs:annotation>
293         <xs:documentation>Número da Carta de Condução.</xs:documentation>
294       </xs:annotation>
295     </xs:enumeration>
296     <xs:enumeration value="NCC">
297       <xs:annotation>
298         <xs:documentation>Número da Segurança Social.</xs:documentation>
299       </xs:annotation>
300     </xs:enumeration>
301     <xs:enumeration value="NP">
302       <xs:annotation>
303         <xs:documentation>Número do Passaporte.</xs:documentation>
304       </xs:annotation>
305     </xs:enumeration>
306   </xs:restriction>
307 </xs:simpleType>
308
309 <!-- IdentificationCodeDescriptionType -->
310 <xs:simpleType name="IdentificationCodeDescriptionType">
311   <xs:restriction base="xs:string"/>
312 </xs:simpleType>
313
314 <!-- IdentificationCodeValueType -->
315 <xs:simpleType name="IdentificationCodeValueType">
316   <xs:annotation>
317     <xs:documentation>
318       Código segundo este sistema de identificação.
319       Por exemplo, se estivermos a armazenar o número de BI, incluir aqui o próprio número.
320     </xs:documentation>
321   </xs:annotation>
322   <xs:restriction base="xs:string"/>
323 </xs:simpleType>
324
325 <!-- IdentificationCodeIssuerType -->
326 <xs:simpleType name="IdentificationCodeIssuerType">
327   <xs:annotation>
328     <xs:documentation>
329       Informação sobre o emissor do código de identificação. Por exemplo: Estado (BI), DGV (
330       carta de condução).
331     </xs:documentation>
332   </xs:annotation>
333   <xs:restriction base="xs:string"/>
334 </xs:simpleType>

```

```

329     </xs:simpleType>
330
331     <!-- IdentificationCodeIssueDateType -->
332     <xs:simpleType name="IdentificationCodeIssueDateType">
333         <xs:restriction base="xs:date"/>
334     </xs:simpleType>
335
336     <!-- IdentificationCodeExpirationDateType -->
337     <xs:simpleType name="IdentificationCodeExpirationDateType">
338         <xs:restriction base="xs:date"/>
339     </xs:simpleType>
340
341     <!-- ContactType -->
342     <xs:complexType name="ContactType">
343         <xs:sequence>
344             <xs:element name="ContactDescription" type="ContactDescriptionType"/>
345             <xs:choice>
346                 <xs:element name="Address" type="AddressType"/>
347                 <xs:element name="Phone" type="PhoneType"/>
348                 <xs:element name="Email" type="EmailType"/>
349                 <xs:element name="URL" type="URLType"/>
350             </xs:choice>
351             <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
352         </xs:sequence>
353     </xs:complexType>
354
355     <!-- ContactDescriptionType -->
356     <xs:simpleType name="ContactDescriptionType">
357         <xs:restriction base="xs:string"/>
358     </xs:simpleType>
359
360     <!-- AddressType -->
361     <xs:complexType name="AddressType">
362         <xs:sequence>
363             <xs:element name="AddressAttentionLine" type="AddressAttentionLineType" minOccurs="0"
364                 maxOccurs="unbounded"/>
365             <xs:element name="AddressLine" type="AddressLineType" maxOccurs="unbounded"/>
366             <xs:element name="AddressPostalCode" type="PostalCodeType"/>
367             <xs:element name="AddressCity" type="CityType"/>
368             <xs:element name="AddressCountry" type="CountryType"/>
369             <xs:element name="Note" type="NoteType" minOccurs="0" maxOccurs="unbounded"/>
370         </xs:sequence>
371     </xs:complexType>
372
373     <!-- PhoneType -->
374     <xs:complexType name="PhoneType">
375         <xs:sequence>
376             <xs:element name="PhoneCountryCode" type="PhoneCountryCodeType" minOccurs="0"/>
377             <xs:element name="PhoneNumber" type="PhoneNumberType"/>
378             <xs:element name="PhoneNumberExtension" type="PhoneNumberExtensionType" minOccurs="0"/>
379         </xs:sequence>
380     </xs:complexType>
381
382     <!-- EmailType -->
383     <xs:simpleType name="EmailType">
384         <xs:restriction base="xs:string">
385             <xs:pattern value="\w@\w\.\w"/>
386         </xs:restriction>
387     </xs:simpleType>
388
389     <!-- URLType -->
390     <xs:simpleType name="URLType">
391         <xs:restriction base="xs:anyURI"/>
392     </xs:simpleType>
393
394     <!-- AddressAttentionLineType -->
395     <xs:simpleType name="AddressAttentionLineType">
396         <xs:restriction base="xs:string"/>
397     </xs:simpleType>
398
399     <!-- AddressLineType -->
400     <xs:simpleType name="AddressLineType">
401         <xs:restriction base="xs:string"/>
402     </xs:simpleType>
403
404     <!-- AddressPostalCodeType -->
405     <xs:simpleType name="PostalCodeType">
406         <xs:restriction base="xs:string"/>
407     </xs:simpleType>
408
409     <!-- AddressCityType -->
410     <xs:simpleType name="CityType">
411         <xs:restriction base="xs:string"/>
412     </xs:simpleType>
413
414     <!-- CountryType -->
415     <xs:simpleType name="CountryType">
416         <xs:annotation>
417             <xs:documentation>
418                 Identificador de um país usando o código numérico da norma ISO 3166-1.
419             </xs:documentation>
420         </xs:annotation>
421         <xs:restriction base="xs:string">
422             <xs:pattern value="\d\d\d"/>

```

```

422         </xs:restriction>
423     </xs:simpleType>
424
425     <!-- PhoneCountryCode -->
426     <xs:simpleType name="PhoneCountryCodeType">
427         <xs:restriction base="xs:string"/>
428     </xs:simpleType>
429
430     <!-- PhoneNumber -->
431     <xs:simpleType name="PhoneNumberType">
432         <xs:restriction base="xs:string"/>
433     </xs:simpleType>
434
435     <!-- PhoneNumberExtension -->
436     <xs:simpleType name="PhoneNumberExtensionType">
437         <xs:restriction base="xs:string"/>
438     </xs:simpleType>
439
440     <!-- ##### -->
441     <!-- StaffStatusType -->
442     <!-- ##### -->
443     <xs:simpleType name="StaffStatusType">
444         <xs:annotation>
445             <xs:documentation>
446                 Definição do tipo para representar o estado de um funcionário numa instituição.
447             </xs:documentation>
448         </xs:annotation>
449         <xs:restriction base="xs:string">
450             <xs:enumeration value="AC">
451                 <xs:annotation>
452                     <xs:documentation>Activo / Active.</xs:documentation>
453                 </xs:annotation>
454             </xs:enumeration>
455             <xs:enumeration value="NA">
456                 <xs:annotation>
457                     <xs:documentation>Não Activo / Not Active.</xs:documentation>
458                 </xs:annotation>
459             </xs:enumeration>
460             <xs:enumeration value="RE">
461                 <xs:annotation>
462                     <xs:documentation>Reformado / Retired.</xs:documentation>
463                 </xs:annotation>
464             </xs:enumeration>
465         </xs:restriction>
466     </xs:simpleType>
467
468     <!-- ##### -->
469     <!-- StaffCategoryType -->
470     <!-- ##### -->
471     <xs:simpleType name="StaffCategoryType">
472         <xs:annotation>
473             <xs:documentation>
474                 Definição do tipo para representar a categoria de um funcionário.
475             </xs:documentation>
476         </xs:annotation>
477         <xs:restriction base="xs:string">
478             <xs:enumeration value="AT">
479                 <xs:annotation>
480                     <xs:documentation>Assistente Estagiário / Assistant Trainee.</xs:documentation>
481                 </xs:annotation>
482             </xs:enumeration>
483             <xs:enumeration value="AS">
484                 <xs:annotation>
485                     <xs:documentation>Assistente / Assistant.</xs:documentation>
486                 </xs:annotation>
487             </xs:enumeration>
488             <xs:enumeration value="RS">
489                 <xs:annotation>
490                     <xs:documentation>Bolsheiro Investigação / Research Scholarship.</
491                     xs:documentation>
492                 </xs:annotation>
493             </xs:enumeration>
494             <xs:enumeration value="SC">
495                 <xs:annotation>
496                     <xs:documentation>Coordenador Secção / Section Coordinator.</xs:documentation>
497                 </xs:annotation>
498             </xs:enumeration>
499             <xs:enumeration value="SD">
500                 <xs:annotation>
501                     <xs:documentation>Director Serviço / Service Director.</xs:documentation>
502                 </xs:annotation>
503             </xs:enumeration>
504             <xs:enumeration value="LD">
505                 <xs:annotation>
506                     <xs:documentation>Director Licenciatura / Licentiateship Degree Director.</
507                     xs:documentation>
508                 </xs:annotation>
509             </xs:enumeration>
510             <xs:enumeration value="MD">
511                 <xs:annotation>
512                     <xs:documentation>Director Mestrado / MSc Degree Director.</xs:documentation>
513                 </xs:annotation>
514             </xs:enumeration>
515         </xs:restriction>
516     </xs:simpleType>

```

```

513         <xs:enumeration value="ID">
514             <xs:annotation>
515                 <xs:documentation>Director Faculdade / Institution Director.</xs:documentation>
516             </xs:annotation>
517         </xs:enumeration>
518         <xs:enumeration value="SP">
519             <xs:annotation>
520                 <xs:documentation>Presidente Conselho Científico / Scientific Committee
521                     President.</xs:documentation>
522             </xs:annotation>
523         </xs:enumeration>
524         <xs:enumeration value="PP">
525             <xs:annotation>
526                 <xs:documentation>Presidente Conselho Pedagógico / Pedagogic Committee
527                     President.</xs:documentation>
528             </xs:annotation>
529         </xs:enumeration>
530         <xs:enumeration value="DP">
531             <xs:annotation>
532                 <xs:documentation>Presidente Departamento / Department President.</
533                     xs:documentation>
534             </xs:annotation>
535         </xs:enumeration>
536         <xs:enumeration value="AP">
537             <xs:annotation>
538                 <xs:documentation>Professor Associado / Associate Professor.</xs:documentation>
539             </xs:annotation>
540         </xs:enumeration>
541         <xs:enumeration value="XP">
542             <xs:annotation>
543                 <xs:documentation>Professor Auxiliar / Auxiliary Professor.</xs:documentation>
544             </xs:annotation>
545         </xs:enumeration>
546         <xs:enumeration value="CP">
547             <xs:annotation>
548                 <xs:documentation>Professor Catedrático / Cathedratc Professor.</
549                     xs:documentation>
550             </xs:annotation>
551         </xs:enumeration>
552     </xs:restriction>
553 </xs:simpleType>
554
555 <!-- ##### -->
556 <!-- NoteType -->
557 <!-- ##### -->
558 <xs:simpleType name="NoteType">
559     <xs:annotation>
560         <xs:documentation>
561             Tipo que representa um elemento genérico para a definição de anotações.
562         </xs:documentation>
563     </xs:annotation>
564     <xs:restriction base="xs:string"/>
565 </xs:simpleType>
566 </xs:schema>

```


Apêndice E

Difusão de Notícias da Universidade

E.1 Especificação do Elemento `NewsItem`

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <xs:schema targetNamespace="http://schemas.up.pt/News" xmlns="http://schemas.up.pt/News" xmlns:xs="http://www.w3.org
  /2001/XMLSchema" elementFormDefault="qualified" attributeFormDefault="unqualified">
3
4     <!-- ##### -->
5     <!-- NewsItem -->
6     <!-- ##### -->
7     <xs:element name="NewsItem" type="NewsItemType"/>
8
9     <!-- ##### -->
10    <!-- NewsItemsType -->
11    <!-- ##### -->
12    <xs:complexType name="NewsItemType">
13        <xs:annotation>
14            <xs:documentation>
15                Item noticioso.
16            </xs:documentation>
17        </xs:annotation>
18        <xs:sequence>
19            <xs:element name="NewsItemIdentification" type="NewsItemIdentificationType">
20                <xs:annotation>
21                    <xs:documentation>
22                        Identificação do item noticioso.
23                    </xs:documentation>
24                </xs:annotation>
25            </xs:element>
26            <xs:element name="Title" type="TitleType">
27                <xs:annotation>
28                    <xs:documentation>
29                        Título da notícia.
30                    </xs:documentation>
31                </xs:annotation>
32            </xs:element>
33            <xs:element name="Date" type="DateType">
34                <xs:annotation>
35                    <xs:documentation>
36                        Data de publicação da notícia.
37                    </xs:documentation>
38                </xs:annotation>
39            </xs:element>
40            <xs:element name="ExpirationDate" type="ExpirationDateType" minOccurs="0">
41                <xs:annotation>
42                    <xs:documentation>
43                        Data de expiração da notícia.
44                    </xs:documentation>
45                </xs:annotation>
46            </xs:element>
47            <xs:element name="Description" type="DescriptionType">
48                <xs:annotation>
49                    <xs:documentation>
50                        Descrição/corpo da notícia.
51                    </xs:documentation>
52                </xs:annotation>
53            </xs:element>
54            <xs:element name="Author" type="AuthorType" minOccurs="0">
55                <xs:annotation>
56                    <xs:documentation>
57                        Autor/responsável pela notícia.
58                    </xs:documentation>
59                </xs:annotation>
60            </xs:element>
61            <xs:element name="Category" type="CategoryType" minOccurs="0" maxOccurs="unbounded">
```

```

62         <xs:annotation>
63             <xs:documentation>
64                 Categoria(s) da notícia.
65             </xs:documentation>
66         </xs:annotation>
67     </xs:element>
68     <xs:element name="URL" type="URLType" minOccurs="0">
69         <xs:annotation>
70             <xs:documentation>
71                 Endereço Web permanente para a notícia.
72             </xs:documentation>
73         </xs:annotation>
74     </xs:element>
75 </xs:sequence>
76 </xs:complexType>
77
78 <!-- ##### -->
79 <!-- NewsItemIdentificationType -->
80 <!-- ##### -->
81 <xs:complexType name="NewsItemIdentificationType">
82     <xs:sequence>
83         <xs:element name="CountryCode" type="CountryCodeType">
84             <xs:annotation>
85                 <xs:documentation>
86                     Código do país.
87                 </xs:documentation>
88             </xs:annotation>
89         </xs:element>
90         <xs:element name="InstitutionCode" type="InstitutionCodeType">
91             <xs:annotation>
92                 <xs:documentation>
93                     Código da instituição.
94                 </xs:documentation>
95             </xs:annotation>
96         </xs:element>
97         <xs:element name="NewsItemCode" type="NewsItemCodeType">
98             <xs:annotation>
99                 <xs:documentation>
100                     Código da notícia.
101                 </xs:documentation>
102             </xs:annotation>
103         </xs:element>
104     </xs:sequence>
105 </xs:complexType>
106
107 <!-- CountryCodeType -->
108 <xs:simpleType name="CountryCodeType">
109     <xs:annotation>
110         <xs:documentation>
111             Identificador de um país usando o código numérico da norma ISO 3166-1. Exemplo:
112                 Portugal = 620.
113         </xs:documentation>
114     </xs:annotation>
115     <xs:restriction base="xs:string">
116         <xs:pattern value="\d\d"/>
117     </xs:restriction>
118 </xs:simpleType>
119
120 <!-- InstitutionCodeType -->
121 <xs:simpleType name="InstitutionCodeType">
122     <xs:annotation>
123         <xs:documentation>
124             Código da instituição no contexto do país. O sistema usado deve ser definido para cada
125             país.
126         </xs:documentation>
127     </xs:annotation>
128     <xs:restriction base="xs:string"/>
129 </xs:simpleType>
130
131 <!-- NewsItemCodeType -->
132 <xs:simpleType name="NewsItemCodeType">
133     <xs:annotation>
134         <xs:documentation>
135             Identificador da notícia, definido pela instituição.
136         </xs:documentation>
137     </xs:annotation>
138     <xs:restriction base="xs:string"/>
139 </xs:simpleType>
140
141 <!-- TitleType -->
142 <xs:simpleType name="TitleType">
143     <xs:restriction base="xs:string"/>
144 </xs:simpleType>
145
146 <!-- ExpirationDateType -->
147 <xs:simpleType name="ExpirationDateType">
148     <xs:restriction base="xs:dateTime"/>
149 </xs:simpleType>
150
151 <!-- DateType -->
152 <xs:simpleType name="DateType">
153     <xs:restriction base="xs:dateTime"/>
154 </xs:simpleType>

```

```
154      <!-- DescriptionType -->
155      <xs:simpleType name="DescriptionType">
156        <xs:restriction base="xs:string"/>
157      </xs:simpleType>
158
159      <!-- AuthorType -->
160      <xs:simpleType name="AuthorType">
161        <xs:restriction base="xs:string"/>
162      </xs:simpleType>
163
164      <!-- CategoryType -->
165      <xs:simpleType name="CategoryType">
166        <xs:restriction base="xs:string"/>
167      </xs:simpleType>
168
169      <!-- URLType -->
170      <xs:simpleType name="URLType">
171        <xs:restriction base="xs:anyURI"/>
172      </xs:simpleType>
173    </xs:schema>
```